LOCAL SYMMETRY AND POLARIZATION IN RELAXOR-BASED FERROELECTRIC CRYSTALS

BY

KYOU-HYUN KIM

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Materials Science and Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2013

Urbana, Illinois

Doctoral Committee:

Professor Jian-Min Zuo, Chair
Professor David A. Payne
Assistant Professor Shen Dillon
Assistant Professor Lane W. Martin

ABSTRACT

Relaxor-ferroelectric single crystals, such as (1-x)$Pb(Mg_{1/3}Nb_{2/3})O_3$-x$PbTiO_3$ (PMN-xPT), have the potential to transform technologies of medical imaging, actuation, and sensors, due to their extraordinary high piezoelectric effect. It has been suggested that polarization rotation driven by external electric fields is responsible for the large piezoelectric response. Polarization rotation is accompanied by a change in crystal symmetry and/or orientation. However, the nature of symmetry in PMN-xPT and other relaxor-ferroelectric crystals, despite extensive study by x-ray and neutron diffraction, is still controversial. Extensive studies have suggested the crystal symmetry varies on the nanoscopic scale and understanding the nature of local symmetry and its variations is thus critical to correlate the structure with polarization properties.

In this thesis, the symmetry is measured based on the size of the crystal volume and the volume position in PMN-xPT single crystals. The study is enabled by the use of probes of different length scales to examine the symmetry of PMN-xPT single crystals. Local symmetry recorded by different probes is measured using quantitative convergent beam electron diffraction analysis (CBED) and transmission electron microscopy (TEM). Symmetry in CBED patterns is correlated with polarization direction with help of simulations. Furthermore, local symmetry fluctuation is observed using a new CBED method developed during this research. The technique enables the quantification of local symmetry development.

ACKNOWLEDGEMENT

I acknowledge and extend my deepest gratitude to many people who have continually supported me. I appreciate my advisor, Prof. Jian-Min (Jim) Zuo for his active guidance and technical supports. This project would not have been possible without his guidance and persistent help. Jim has continually motivated me with new adventurous ideas and new techniques, which have made my graduate study fun, exciting and sometimes challenging.

I would also like to thank Prof. David A. Payne for his considerable help and comments on this thesis with his knowledge of electro-ceramics. Prof. Payne has contributed to the results described in a number of chapters and has taught me like his own student since I started this project. I would also like to thank Prof. Lane W. Martin and Prof. Shen Dillon who have served as my doctoral committee members.

Finally, I owe my greatest gratitude to my family for their support. My wife and my son have been a great pleasure for me during my study at University of Illinois at Urbana-Champaign. My parents have always supported me, and their support cannot be described sufficiently by any words.

Dedicated to the people I owe most,

My parents, who have always supported me and my family

Yeawon, who has always stood for me

Taeyeon (David), who has taught us love

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Piezoelectricity

Piezoelectricity was originally discovered in 1880 by the brothers Jacques and Pierre Curie [1, 2]. The two brothers observed that some materials could develop an electric potential when an external pressure was applied along specific directions. Piezoelectric effects were found in materials such as quartz (SiO_2), Rochelle salt (also known as Seingette salt), tourmaline, zinc blende, sodium chlorate, boracite, calamine, topaz, tartaric acid, and cane sugar. The Curie brothers also measured the magnitude of the piezoelectric effect in quartz, and their measurements showed that the piezoelectric induced charge was proportional to the applied pressure. By the end of 1881, the Curie brothers had discovered the existence of the converse effect and demonstrated that the magnitude of the piezoelectric constants of quartz were the same for the direct and converse effects.

A large volume of studies soon followed in search of other materials for their piezoelectric effects and also for their applications. For example, the technology of oscillator or resonator was invented using quartz in 1921, and the quartz crystal oscillator was soon used for making precise clocks [3-6]. Quartz was also used as precise frequency reference crystals for radio transmitters. Piezoelectric effects were also discovered in a group of materials with perovskite structure such as barium titanate ($BaTiO_3$), lead zirconate ($PbZrO_3$), and lead titanate ($PbTiO_3$). These materials were used in microphones and ultrasonic devices. Lead zirconate-lead titanate system ($Pb(Zr_{1-x}Ti_{1-x})O_3$ (PZT)) was introduced in the early 1950s by Jaffe and his coworkers [7, 8]. They observed that the PZT system has a nearly vertical and temperature-independent phase boundary between the R phase (*3m*) and the T phase (*4mm*), which is called the morphotropic phase boundary (MPB). The physical properties of PZT are sensitive to the Zr/Ti ratio. Lead zirconate titanate (PZT, $Pb(Zr_xTi_{1-x})O_3$) is the standard piezoelectric material used for medical imaging. Piezoelectric materials change their shape on the order of sub-nanometers when an electric field is applied. This property is used for the actuation of precise probe control in scanning tunneling microscopes [9].

Table 1.1: Crystal classes for polar and piezoelectric material.

Crystal systems	32 point groups						
Triclinic	1**	$\bar{1}$					
Monoclinic	2**	m**	$\frac{2}{m}$				
Orthorhombic	222*	$2mm$**	mmm				
Tetragonal	4**	$\bar{4}$*	$\frac{4}{m}$	422*	$4mm$**	$\bar{4}2m$*	
Trigonal	3**	$\bar{3}$	322*	$3m$**	$\bar{3}m$		
Hexagonal	6**	$\bar{6}$*	$\frac{6}{m}$	622*	$6mm$**	$\bar{6}2m$*	$\frac{6}{m}mm$
Cubic	23*	$\frac{2}{m}3$	432	$\bar{4}3m$*	$\frac{4}{m}3\frac{2}{m}$		

('*' indicates piezoelectric systems, and '**' indicates polar crystal systems having a spontaneous polarization)

In piezoelectric crystals, piezoelectricity is only observed when the crystal does not possess a center of symmetry [10]. Table 1.1 lists the 32 crystallographic point groups. 11 classes of these possess the center of symmetry and are thus non-piezoelectric crystal. Among the remaining 21 non-centrosymmetric classes, piezoelectric effect can be observed in 20 point groups of *1*, *2*, *m*, *222*, *2mm*, *4*, $\bar{4}$, *422*, *4mm*, $\bar{4}2m$, *3*, *322*, *3m*, *6*, $\bar{6}$, *622*, *6mm*, $\bar{6}2m$, *23*, and $\bar{4}3m$. Of these 20 point groups, 10 point groups of *1*, *2*, *m*, *2mm*, *4*, *4mm*, *3*, *3m*, *6*, and *6mm* are grouped into polar crystals because they have a spontaneous polarization. In the polar state, the center of positive charge does not coincide with the center of negative charge. Thus, the polar crystals possess a permanent dipole. The spontaneous polarization (**P**$_S$) is dependent on temperature, which is called *pyroelectricity*. *Ferroelectricity* is a special case of pyroelectricity, in which the **P**$_S$ direction can be switched by an electric field. It is therefore all ferroelectric crystals are pyroelectric and all pyroelectric crystals are piezoelectric, but not vice versa. The difference between a piezoelectric and a ferroelectric crystal is schematically represented in Fig. 1.1.

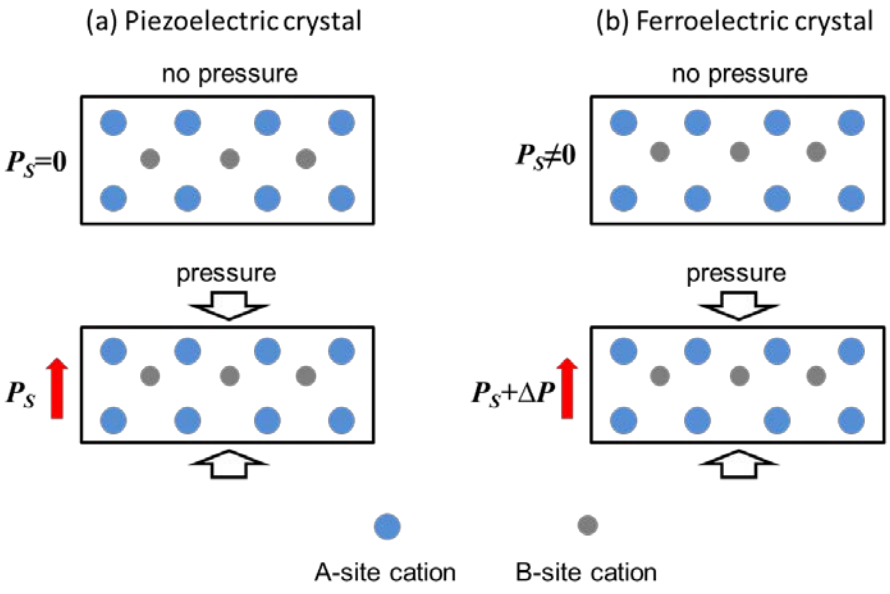


Figure 1.1: (a) Unstressed piezoelectric material (***Ps***=0) exhibits induced polarization under pressure. (b) Ferroelectric crystal shows $\boldsymbol{P}_S$ in the unstressed state, and the applied stress develops the polarization change by Δ***P***.

Piezoelectric materials based on the formula of ABO_3 with the perovskite structure are the focus of research in past several decades [10]. The ideal perovskite has a primitive cubic structure of $Pm\bar{3}m$ (or $Pm3m$) symmetry (Fig. 1.2). The unit cell origin can be selected on either the A-site or B-site cation. Figure 1.2 shows that the origin of perovskite unit cell is selected on the A-site cation. In the perovskite structure, the A-cations are surrounded by 12 oxygens, and B-cations are coordinated by 6 oxygens forming BO_6^{8-} octahedron as shown in Fig. 1.2. The ideal perovskite structure has centric-symmetry and then no spontaneous polarization due to the absence of separation between negative and positive charge. The symmetry reduction occurs when the B-site cation spontaneous displaces relative to the A-caions and O ions [10, 11]. A separation of positive and negative charges then develops accompanied by symmetry reduction, resulting in a permanent $\mathbf{P}_\mathrm{S}$ dipole. The permanent dipole is restricted by symmetry. For example, the B-cation displacement along [001] in an ideal perovskite reduces the cubic symmetry to the tetragonal symmetry. Thus, $\mathbf{P}_\mathrm{S}$ is restricted along [001] in structures with tetragonal (T) symmetry. For rhombohedral structure, the B-cation moves along <111>, leading to the $\mathbf{P}_\mathrm{S}$ along [111]. By applying an electric field, the polarization direction of ferroelectric crystal may be switched (or re-oriented) into the permissible directions allowed by symmetry [10, 11]. Figure 1.3 shows the phase transformation from the ideal perovskite structure to rhombohedral or tetragonal structure with different polarization directions.

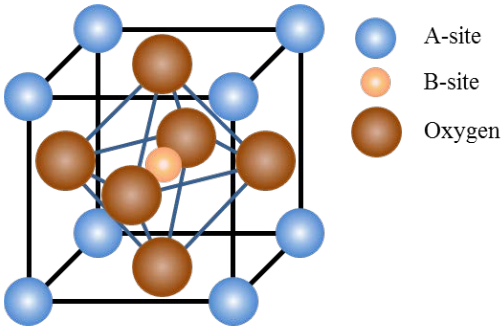


Figure 1.2: Ideal perovskite structure of ABO_3.

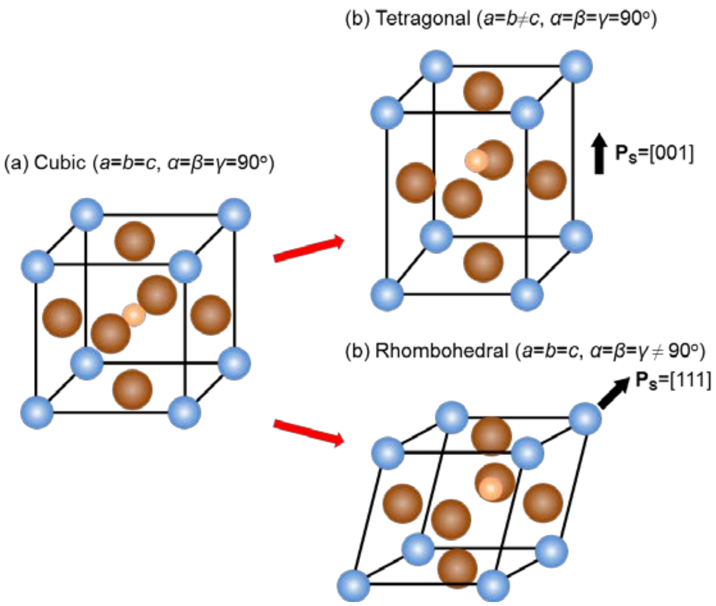
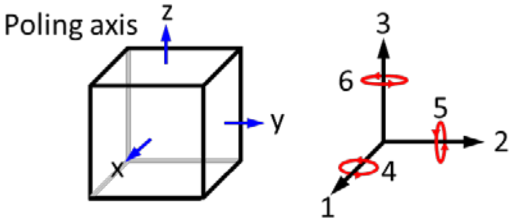


Figure1.3: Symmetry changes from (a) the ideal perovskite structure (cubic) to (b) tetragonal or (c) rhombohedral structure. The spontaneous polarization direction is determined by the symmetry of crystal.

Piezoelectric properties are described by several constants, represented by different symbols. Table 1.2 shows the basic constants, their symbols and terminologies in piezoelectricity.

Table 1.2: Basic piezoelectric constants according to refs [10, 11]



K_3^S	**Relative dielectric constant** All strains to the material are constant, or mechanical deformation is blocked in any direction Electrodes are perpendicular to the axis 3	
K_1^T	**Relative dielectric constant** All stresses on the material or no external foces Electrodes are perpendicular to the axis 1	
k_P	**Electromichanical coupling factor** Stress or strain is equal in all directions perpendicular to the axis 3. This constant is only used when electrodes are perpendicular to the axis 3.	
k_{15}	**Electromichanical coupling factor** Stress or strain is in shear form around the axis 2 Electrodes are perpendicular to the axis 1	$k_{ij} = \sqrt{\frac{\text{Electrical (or mechanical)energy converted}}{\text{Mechanical (or electrical)energy applied}}}$
d_h	**Piezoelectric charge constant (m/V or C/N):** polarization generated per unit of mechanical stress applied to a piezoelectric material Hydrostatic stress or stress is equally applied in all directions. Electrodes are perpendicular to the axis 3.	
d_{33}	**Piezoelectric charge constant (m/V or C/N)** Applied stress or induced strain in direction 3 (parallel to the polarized direction) Electrodes are perpendicular to the axis 3 *d_{33} is also known as *piezoelectric modulus*.	$d_{ij} = \frac{\text{Strain}}{\text{Applied field}} = \frac{\text{Charge density}}{\text{Applied stress}}$
g_{15}	**Piezoelectric voltage constant (Vm/N)** Applied stress or piezoelectrically induced strain is in 3-direction Electrodes are perpendicular to the axis 3	$g_{ij} = \frac{\text{Strain}}{\text{Applied charge density}}$ $= \frac{\text{Strain}}{\text{Applied mechanical stress}}$
S_{11}^E	**Elastic constant (1/Pa)** Measured in closed circuit Stress or strain in direction 1 Stress or strain in direction 1	$S_{ij} = \frac{\text{Strain}}{\text{Stress}}$
S_{11}^D	**Elastic constant (1/Pa)** Measured in open circuit Stress or strain in direction 1 Stress or strain in direction 1	$S_{ij} = \frac{\text{Strain}}{\text{Stress}}$

1.2 Relaxor-based ferroelectric crystals

Relaxor-based ferroelectric (or simply relaxor-ferroelectric) crystals are solid solutions between relaxors and ferroelectric crystals. For example, relaxor-ferroelectric (1-x)$Pb(Mg_{1/3}Nb_{2/3})O_3$-x$PbTiO_3$ (PMN-xPT) is a solid solution between the relaxor PMN and the ferroelectric PT. PMN-PT exhibits the characteristics of relaxor or ferroelectric dependent on the PT content. Relaxor-based ferroelectric single crystals have attracted much research attention and application interest after Park et al. demonstrated giant piezoelectric coefficients in the single crystals of PMN-xPT [12-14] (Table 1.3).

Table 1.3: Piezoelectric properties of common piezoelectric and relaxor-ferroelectric crystals

Materials	Crystal structure	S_{11}^{E} (1/Pa)	S_{11}^{E} (1/Pa)	d_{31}^{E} (pm/V)	d_{33}^{E} (pm/V)	k_{31}
$BaTiO_3$ [15]	P4mm	8.05	15.7	-34.5	85.6	0.315
PZT [16]	Pm3m R3m P4mm	15~17	19~22	-140~ -280	300~ 600	0.28~0.48
PMN-PT [17] x$Pb(Mg_{1/3}Nb_{2/3})O_3$ -(1-x)$PbTiO_3$	Pm3m R3m P4mm	50~70	40~120	-800~ -1500	1500~ 3000	~0.94
PZN-PT [17] x$Pb(Mg_{1/3}Nb_{2/3})O_3$ -(1-x)$PbTiO_3$	Pm3m R3m P4mm	80~90	110~145	-950~ -1600	2000~ 2900	~0.58

Relaxors, such as PMN or PZN, contain multiple B-cations and exhibit distinguished properties from normal ferroelectrics such as PT. The dielectric properties of normal ferroelectrics do not vary greatly with frequency. However, relaxors show a pronounced change in permittivity with frequency at temperatures near the Curie point. In addition, relaxors and ferroelectrics differ in their phase transformation behavior. Consider the Gibbs function G of a system, the change in G for the system can be written

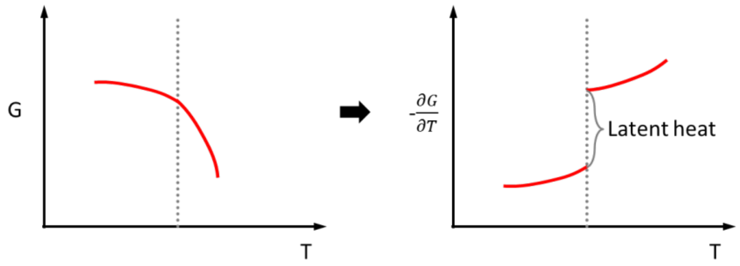
$$dG = -SdT + VdP \qquad (1.\text{--})$$

where S, V and P are entropy, volume and pressure, respectively. The entropy is then

$$S = -\left(\frac{\partial G}{\partial T}\right)_P \qquad (1.\text{--})$$

The derivative of $\frac{\partial G}{\partial T}$ at a constant pressure can be continuous or discontinuous. The discontinuity indicates an abrupt change in entropy (or latent heat) at a certain temperature. This is the characteristic of a 'first-order (1st-order)' transition (Fig. 1.4). For a 'second-order (2nd-order)' transition, the Gibbs function continuously changes with temperature. In contrast to the 1st-oder transition, the first order derivative of $\frac{\partial G}{\partial T}$ also continuously changes with temperature as illustrates in Fig. 1.4.

(a) 1st-order phase transformation



(a) 2nd-order phase transformation

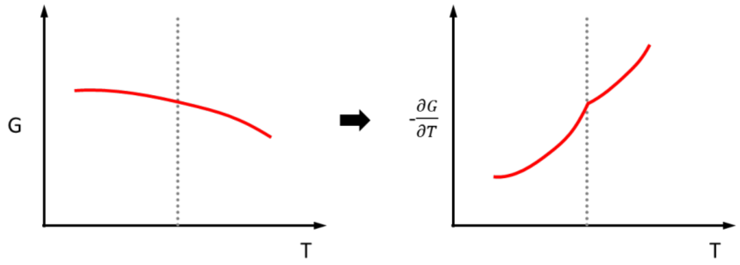


Figure 1.4: Free energy (G) changes for (a) the first-order and (b) the second order phase transition as a function of temperature (T)

In general, a ferroelectric phase transformation is described by a 1st-order phase transition behavior. Some complex perovskites with multiple B-site cations, however, show 2nd-order phase transition behavior, and the dielectric constant of these complex perovskites exhibit a continuous change with temperature.

1.3 Morphotropic phase boundary (MPB)

The term 'morphotropic' originally refers to a chemically driven change in morphology or phase [18, 19]. However, the term of morphotropic phase boundary (MPB) frequently indicates a specific boundary that separates regions between rhombohedral and tetragonal symmetry by varying the composition [7, 8, 11]. That is, the MPB boundary refers to a phase boundary that is largely determined by composition.

Figure 1.5 shows the phase diagram of PMN-xPT, a relaxor-based ferroelectric. The phase diagram for PMN-xPT was first studied by H. Ouchi et al [20-22]. According to Ouchi, the MPB is located around the composition at x=41% for PMN-xPT. Since then, the MPB composition has been revised. The location of MPB is now considered to be x=28~35% [23, 24]. According to published x-T phase diagrams for PMN-xPT (Fig. 1.5), the prototypic high-temperature (HT) phase of cubic symmetry (*Pm3m*) spontaneously distorts to rhombohedral (R) *R3m* symmetry upon cooling at low x, or tetragonal (T) *P4mm* symmetry at higher x, in which $\mathbf{P}_S$ and ε_S are constrained to the cubic (noted by the subscript 'C') $[111]_C$ and $[001]_C$ directions for the R and T phases, respectively. The R and T phases are initially separated by the almost vertical MPB [25, 26]. The MPB of PMN-xPT and PZN-xPT is characterized by a narrow composition region where the R and T phases meet. This phase boundary composition has attracted much attention because displacements maximize as the lattice softens and transforms, giving rise to a large enhancement in piezoelectric properties [13, 14].

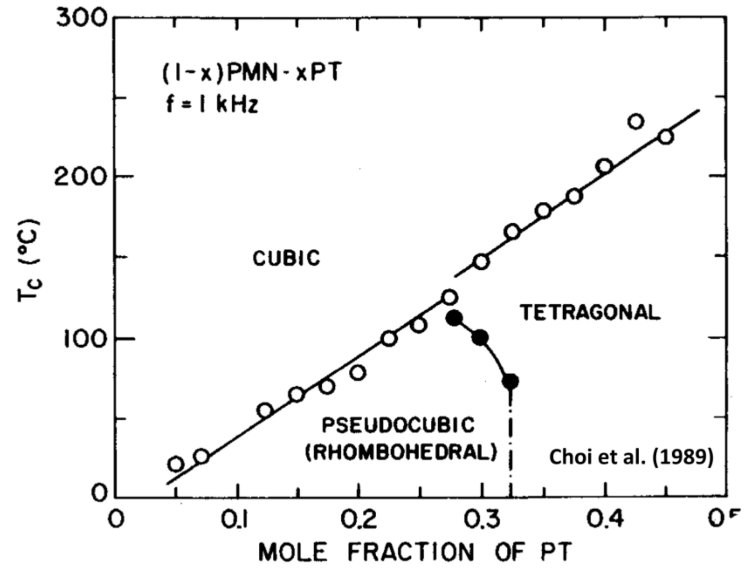


Figure 1.5: Phase diagram of PMN-xPT system plotted from the dielectric data published in the ref [23]

1.3.1 Phase coexistence near the morphotropic phase boundary

A large body of literature focuses on the MPB region of PZT with Zr/Ti=58:42 ~ 48:52 [7, 11, 27-31]. At this composition, it shows the highest piezoelectric properties in the PZT system. The origin of high piezoelectric properties of PZT in MPB was first believed to originate from coexistence of rhombohedral and tetragonal phases [30, 32-41].

Flux growth was initially attempted for the growth of PZT ceramics. [39, 42, 43]. A PZT singe crystal, however, was hardly achieved by the flux-growth method. For these reasons, the MPB region in PZT polycrystals was considered to be a mixed region with R and T phases. It was long believed that the coexistence of rhombohedral and tetragonal phases gives rise to stronger piezoelectric response due to a greater number of possible polar vectors relative to the poling direction [11, 44].

A controversy, however, arose when the relaxor-based ferroelectric single crystals such as PMN-PT and PZN-PT were introduced. These single crystals show enhanced piezoelectric responses along poling directions that do not correspond to the polar axis [12-14, 25]. The highest piezoelectric properties are observed in $[001]_C$ poled single crystals with the composition inside the rhombohedral phase region that is close to the MPB. For the two-phase coexistence model, however, the specific poling direction ($[001]_C$) is not necessary due to a great number of polar vectors.

1.3.2 Polarization rotation model

Fu and Cohen proposed the polarization rotation mechanism for high piezoelectric responses in piezoelectric single crystals [45]. A major issue, however, was that the R phase cannot directly transform into the T phase due to symmetry restriction [24-26, 41, 45-50]. Around the same time, new phases were found in the MPB region by means of high resolution x-ray, optical microscopy and neutron diffraction [24-26, 41, 46-48, 51-53]. The new phases are said to be monoclinic (M) symmetry with space groups of *Cm* or *Pm* according to the notations of Vanderbilt and Cohen [48].

The c_M (*c*-axis) of monoclinic (*Cm*) is almost parallel to $[001]_C$ with a small tilt angle of β, and the a_M (*a*-axis) and b_M (*b*-axis) are rotated 45° about $[001]_C$ [25, 26, 46]. The monoclinic *Cm* symmetry has a mirror plane normal to the $[010]_M$ (=$[1\bar{1}0]_C$). In the polarization rotation model, polarization direction lies between $[111]_R$ (=$[111]_C$) and $[001]_T$ (=$[001]_C$) [45, 46]. Therefore, the polar vector is crystallographically free to rotate from R ($\mathbf{P_S}$=$[111]_C$) to T ($\mathbf{P_S}$= $[001]_C$) within *Cm* acting as the bridging phase. For the monoclinic *Pm*, the c_M is almost parallel to $[001]_C$ with a small tilt angle of β, and the a_M and b_M lie along $[100]_C$ and $[010]_C$, respectively [25, 54]. The monoclinic *Pm* has a mirror plane along the $[010]_M$ (=$[010]_C$). This mirror plane then acts as a structural bridge for polarization rotation between $[101]_O$ and $[001]_T$ [25]. The intermediate phases of monoclinic *Cm* and *Pm*, therefore, define the polar rotation paths from R to T phase.

1.3.2.1 Nature of monoclinic symmetry

The polarization rotation model suggests the ease of polarization rotation from rhombohedral to tetragonal phase via the intermediate monoclinic phases. The nature of monoclinic symmetry, however, is still being debated. First, the amounts of monoclinic phases detected in single crystals are too small to facilitate the polarization rotation from the rhombohedral phase to the tetragonal phase [55]. Second, several research groups reported that the monoclinic phase observed at the macroscopic scale does not reflect the real local structure [56-59]. That is, the local structure is different from the symmetry at the macroscopic scale. Third, it was observed that the phase transition from R phase to T phase is continuous, i.e., a 2^{nd}-order phase transition [17, 47]. The direct phase transition from R to T, however, is not allowed by symmetry, resulting in the 1^{st}-order transition [50, 59, 60]. Two models of adaptive phase or monoclinic distortion have been proposed in order to explain the monoclinic phases observed in the MPB region.

1.3.2.1.1 Adaptive phase model

The adaptive phase model was originally developed for the martensite transformation in steel [61]. The martensitic transformation is a 1st-order solid-state displacive phase transformation (also known as diffusionless transformation) with a homogeneous lattice deformation [62]. The martensite transformation proceeds along a specific direction which provides almost complete accommodation of the crystal-lattice mismatch, resulting in periodic lamellae of two-related orientational variants of the martensitic phase as shown in Fig. 1.6. The habit plane of the martensite plate remains undistorted following the transformation so that the habit plane coincides with the twinning plane of two different phases [62].

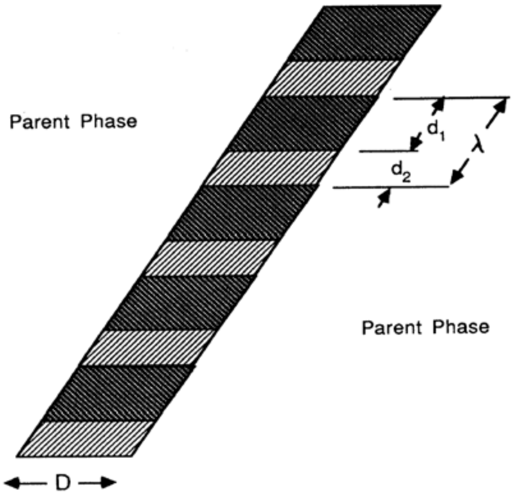


Figure 1.6: Schematic of martensite plates comprised of twin-related lamellae of two orientation variants [61]

Martensite and relaxor-ferroelectrics share many similarities in microstructural aspects. First, a tweed-like microstructure (or nanodomain)[1] is observed in both of the martensite and the

---

[1] The tweed-like contrast has been suggested as evidence for nanodomains even though there was no direct experimental evidence for the polarization directions in these domains. Such evidence is provided in Chapter 7.

relaxor-ferroelectrics [57, 58, 61, 63]. Second, the nanodomains in relaxor-ferroelectrics are considered to form a twin plane like the martensite. Based on these similarities, Viehland and coworkers propose the adaptive phase model for the relaxor-ferroelectrics [57, 58, 63].

The adaptive phase model is a result of averaging over nanodomains for the monoclinic symmetry at macroscopic scale, and R or T nanodomains form a finely twinned structure, leading to a different macroscopic symmetry from the local symmetry. According to the adaptive phase model, the finely twinned rhombohedral nanodomain give rise to the monoclinic M_A (or M_B) while the finely twinned tetragonal nanodomain give monoclinic M_C symmetry. The monoclinic M_A and M_B have the same space group of *Cm*, but have different polarization directions along [*uuv*] ($u \ll v$) and [*uuv*] ($u \gg v$), respectively. The monoclinic M_C has space group *Pm* with a polarization direction of [u0v]. The macroscopic symmetry of monoclinic phases, therefore, is the averaged symmetry over the twinned nanodomains.

1.3.2.1.2 Monoclinic distortion model

Figure 1.7 shows the possible subgroups of a prototype cubic symmetry ($Pm\bar{3}m$) for perovskite that can be induced by a ferroelectric displacement of the B-cation according to Forrester et al [50]. The diagram shows that the space group *Cm* is a subgroup of *R3m* while the space group of *Pm* is a subgroup of *P4mm*. Thus, according to this diagram, the phase transition from R to monoclinic *Cm* is allowed by symmetry while *R3m* and *Pm* are not group-related. This implies that the monoclinic transition from R phase must be discontinuous on the local level, i.e., a 1st-order transition. A number of research groups, however, reported that the PZN-PT system shows a continuous field-induced phase transition from R to M, which contradicts the symmetry argument [47].

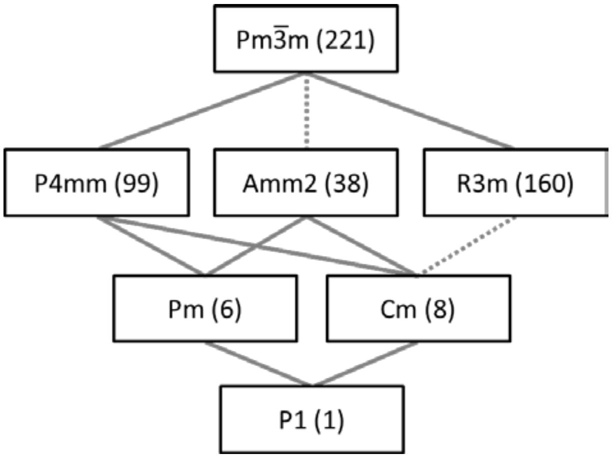


Figure 1.7: The group-subgroup relationships for the $Pm\bar{3}m$ symmetry observed in the perovskite structures. The 1st-order phase transition is represented by the dotted line while the 2nd-order phase transition is represented by the solid line.

Based on the observed continuous phase transition, the proponents of the monoclinic distortion model propose that the observed M phases are not necessarily a true structural phases, but distorted structures induced by an external electric force [59]. That is, the observed M symmetry is simply a result of the piezoelectric response to the applied electric field. The perovskite single crystal deforms under the external electric field, and the deformation itself can lower the symmetry of the crystal without a phase transition. If the phase transition is not involved, the deformation process itself is continuous. Kisi et al. also shows that an infinitesimal electric field along any of the rhombohedral axes ($[100]_R$, $[010]_R$ or $[001]_R$) induces a monoclinic deformation which lowers *R3m* to *Cm* [59]. Thus, what is most important for polarization rotation is not the presence of monoclinic phases but the elastic compliance of PZN-PT single crystals, which will be soft at the MPB composition.

1.4 Motivation

The above discussions show that the M phases (*Cm* and *Pm*) and the roles they play are critical to understand the giant piezoelectric responses in complex ferroelectric single crystals. A large body of literature has been devoted to the study of the exact symmetry of these crystals [24, 25, 41, 46, 48, 49, 51-53, 64, 65]. Never-the-less, debates continue on their exact symmetry. The major issue stems from the different methods for the symmetry study. The symmetry has been investigated using optical microscopy (OM), high resolution XRD, neutron diffraction, and TEM [24, 49, 51, 52, 66, 67], which have different spatial resolutions. The resulting conclusions, therefore, reflect the different scales of measurement. Surprisingly, no studies have been reported on the relationship between the local and the macroscopic symmetry.

Thus, determination of symmetry from local to macroscopic level in the MPB region is critical to settle the above disputes and understand the high piezoelectric performance in perovskite single crystals. In this thesis, the symmetry of PMN-PT will be investigated from the local to the microscopic level using the CBED technique and XRD, respectively. For the symmetry investigation, a newly developed symmetry quantification method will be used together with a scanning electron diffraction technique. These techniques allow for a study of volume-dependent symmetry changes in PMN-PT single crystals as well as locally varying polarization. The results of this thesis provide the missing link between the local symmetry and the macroscopic symmetry, including the polarization changes across nanodomains.

CHAPTER 2

SCANNING CONVERGENT BEAM ELECTRON DIFFRACTION

This chapter describes the scanning convergent beam electron diffraction (SCBED) method and the related issues. The symmetry and domain structure study reported in Ch. 4~7 are based on these techniques described in this chapter. The first two sections introduce the principles of electron probe formation and symmetry quantification using CBED. This is followed by a section describing the experimental issues involved in SCBED. The theory of CBED and its simulation are described in the last section.

2.1 Electron probe formation

2.1.1 Illumination system

Two types of TEMs are used to obtain different electron probes of different sizes for the study described in this thesis. The first is a 2100 Cryo TEM (JEOL, Japan) equipped with a LaB_6 thermionic emission gun (Fig. 2.1(a)). The second type is 2010F-FEG (JEOL, Japan) with a field emission gun (Fig. 2.1(b)). Table 2.1 lists the difference between the two electron guns.

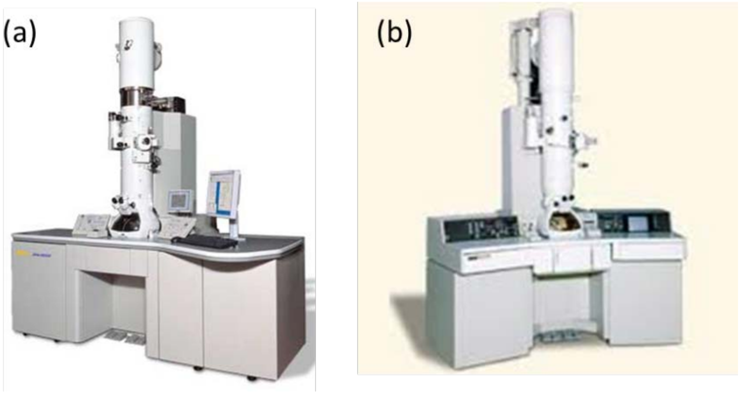


Figure 2.1: (a) 2100 Cryo TEM with a LaB_6 thermionic emission gun and (b) 2010F-FEG TEM with a field emission gun [68]

Table 2.1: Characteristics of thermionic (LaB_6) and field emission gun sources [69, 70]

	LaB_6	Field emission
Work function, ϕ (eV)	2.7	4.5
Operation temperature (K)	1700	300
Current density (A/Cm^2)	25 ~ 100	10^{10}
Beam crossover size (μm)	10	<0.01
Brightness ($A/cm^2 \cdot sr$)	10^6	10^{13}
Energy spread (eV)	1.5	0.3
Gun vacuum (Pa)	10^{-4}	5

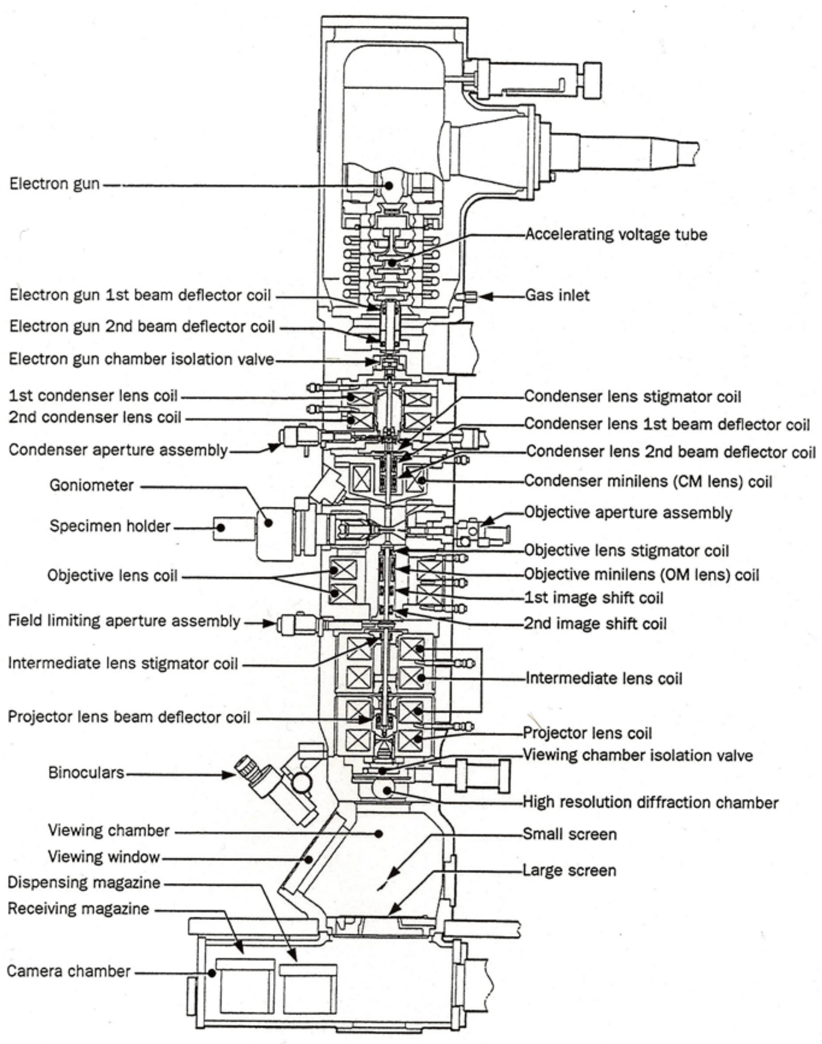


Figure 2.2: Cross section of JEOL 2010F-FEG

The illumination lens system is used in TEM to make a focused or parallel beam on the specimen. The basic illumination system consists of an electron gun, condenser lenses (CL), CL apertures, and deflection coils. Figure 2.2 shows a schematic cross-sectional diagram of JEOL 2010F TEM. The electrons emitted from the gun are accelerated and then imaged by the 1st (CL1), 2nd (CL2) condenser lenses and condenser minilens (CM lens). Figure 2.3 shows the working of these lenses for electron illumination under the different modes. The CL1 demagnifies the electron source image coming from the gun. The size of electron beam crossover after the CL1 increases as the excitation of CL1 lens decreases. The CL2 and CM lenses focus the electron beam onto the sample. The CM lens is fixed according to illumination mode such as TEM, CBED or NBED, while the CL2 lens makes a parallel beam or a focused beam. The parallel beam is used for TEM imaging and selected area diffraction pattern (SADP), and the focused beam is used for scanning TEM (STEM) or microdiffraction such as convergent beam electron diffraction (CBED).

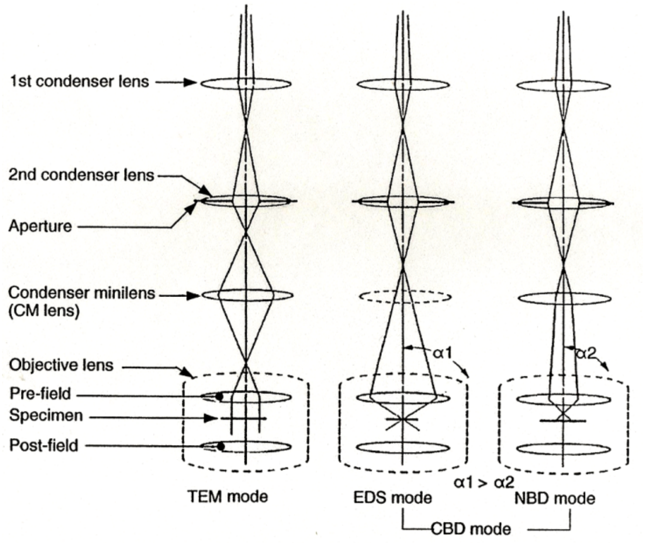


Figure 2.3: Ray diagram of illumination system for TEM, EDS, NBD, and CBED mode

2.1.2 Probe diameter

The probe diameter d_0 on the sample is initially determined by the source image diameter after CL1 (d_s). The source image is broadened by the effects of spherical aberration in the probe-forming-lens (d_{sa}) and diffraction by aperture (d_d). The chromatic aberration (d_c) of probe forming lens and focusing error (d_f) also contribute to the broadening of the probe diameter. The total beam probe diameter can be expressed as following

$$\begin{aligned} d_0^2 &= d_s^2 + d_d^2 + d_{sa}^2 + d_c^2 + d_f^2 \\ &= d_s^2 + \left(\frac{0.6\lambda}{\theta_c}\right)^2 + \left(0.5C_s\theta_c^3\right)^2 + \left(\frac{\Delta E_0}{E_0}C_c\theta_c\right)^2 + \left(2\theta_c\Delta f\right)^2 \end{aligned} \tag{2.1}$$

,where λ is the wave length of the electron beam, θ_c is the beam convergence angle, C_s is spherical aberration coefficient, C_c is a chromatic aberration coefficient, and Δf is a defocus value [71]. The minimum probe size can be obtained by minimizing the sum of all terms in Eq. 2.1 as shown in Fig. 2.4.

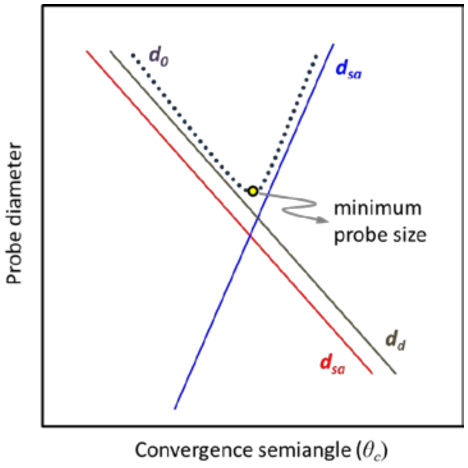


Figure 2.4: Calculations for minimum probe size as a function of convergence semiangle at zero focus error

The size of electron probes in a TEM equipped with a LaB_6 thermionic emission gun is largely determined by the source electron image size. Experimentally, the different lens setting for CL1 gives the different probe size as shown in Fig. 2.4. We used the JEOL 2100 LaB_6 installed in CMM, UIUC, in the CBED mode and changed the demagnification of the condenser 1 lens (spot size). The beam diameter is experimentally defined by the full-width at half-maximum (FWHM) of the electron-intensity in the Gaussian-distribution. Using this approach, we were able to obtain electron probes from 2.7 to 24 nm at FWHM as shown in Fig. 2.5.

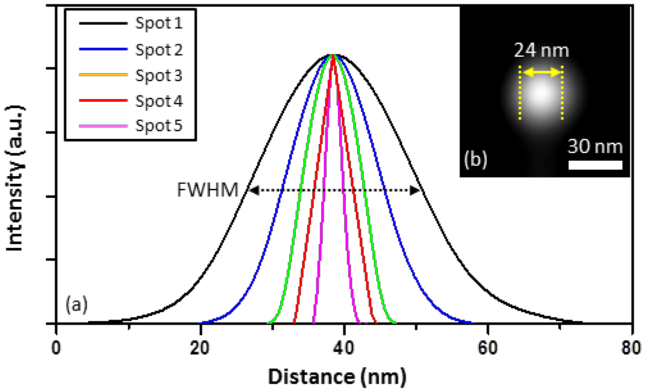


Figure 2.5: Full-width at half-maximum (FWHM) for the probe size determination. (a) Intensity profile of the beam probe from 24 nm (black) to 2.7 nm (pink). The (b) shows the image of 24 nm beam probe.

In a TEM equipped with a field emission gun, the electron illumination inside the probe forming aperture can be considered as fully coherent to a good approximation. The intensity distribution of an electron beam probe at the sample is then given by [71, 72]

$$I(\mathbf{r}) = \Psi_{\mathrm{p}}(\mathbf{r})\Psi_{\mathrm{p}}(\mathbf{r})^{*} \tag{2.2}$$

$$\Psi_{\mathrm{p}}(\mathbf{r}) = F^{T}\left\{A(\mathbf{K_t})\exp[i\chi(\mathbf{K_t})]\right\} \tag{2.3}$$

F indicates a Fourier transformation. The electron beam intensity across an entrance aperture is uniform, and this is known as a "top-hat" intensity profile. The top-hat intensity profile A($\mathbf{K_t}$) is a function of width θ_c of the illumination aperture. The wave-front aberration function $\chi(\mathbf{K_t})$ is given by

$$\chi(\mathbf{K_t}) = \pi(\Delta f\,\lambda K_t^2 + 0.5 C_s \lambda^3 K_t^4) \tag{2.4}$$

Here, c is the probe coordinate. According to Mory *et al.*, the Δf value can be determined by the probe radius containing 70% of the beam intensity [73]. Using Eqs. 2.2 and 2.3, the conditions for optimizing the electron probe size can be calculated as following

$$\theta_c = 1.27 C_s^{-0.25} \lambda^{0.25} \tag{2.5}$$

$$\Delta f = -0.75 C_s^{0.5} \lambda^{0.5} \tag{2.6}$$

The minimum probe diameter containing 70% of the intensity can be then calculated as

$$d(70\%) = 0.66 \lambda^{0.75} C_s^{0.25} \tag{2.7}$$

2.2 Scripting for scanning electron diffraction and symmetry quantification

Scanning electron diffraction technique (SED) is based on a script language pre-installed in Digital Micrograph$^{\mathrm{TM}}$ (DM). DM is software used to acquire, process and analyze images and spectrum data in TEM that was developed by Gatan Inc [74]. The software includes a script language (also known as DM script) that can be used to customize tasks such as TEM operation and image manipulation [75]. An electron microscope can be controlled using DM script by communicating with the host processor built-in inside TEM and with the correct microscope control plug-in installed. Using the DM scripting language, users can create new functionalities that go beyond what is provided by what was initially installed in the DM software. A large body of DM script examples can be found in online libraries [76, 77].

2.2.1 Scanning electron diffraction (SED)

TEM has two sets of deflection coils as shown in Fig. 2.2. The first deflection coils tilt or shift the electrons coming from the gun, while the second deflection coils tilt or shift the beam formed by CL2. Figure 2.6 schematically shows the role of deflection coils to tilt or shift the electron beam. Two deflection coils work together to tilt or shift the electron beam.

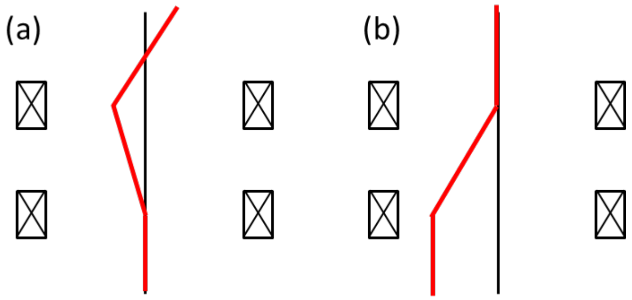


Figure 2.6: Schematic of the actions of deflection coils to (a) tilt and (b) shift

Table 2.2: TEM control commands for scanning electron diffraction

Synopsis	EMGetBeamShift (Number x, Number y)
Description	Retrieve the value of beam shift coil and assign to the number of x and y
Synopsis	EMSetBeamShift (Number x, Number y)
Description	Shift the beam by the amounts of x and y
Synopsis	SSCGainNormalizedBinnedAcquire (Image , Exposure, Binning, top, left, bottom, right)
Description	Acquire an gain normalized SSC frame

The beam shift can be controlled using the DM script language. For scanning electron diffraction (SED), the basic control commands are listed in Table 2.2. The script for SED uses two main commands of (1) EMGetBeamShift and (2) EMSetBeamShift in order to position the electron beam. Diffraction patterns are acquired using the SSCGainNormalizedBinnedAcquire command. The first command, EMGetBeamShift, retrieves the value of beam shift coils and stores the value into the assigned real number x and y. The values of x and y, however, only refer to the setting values of electromagnetic lenses. Thus, the values need to be converted into a real unit such as nanometer or micrometer.

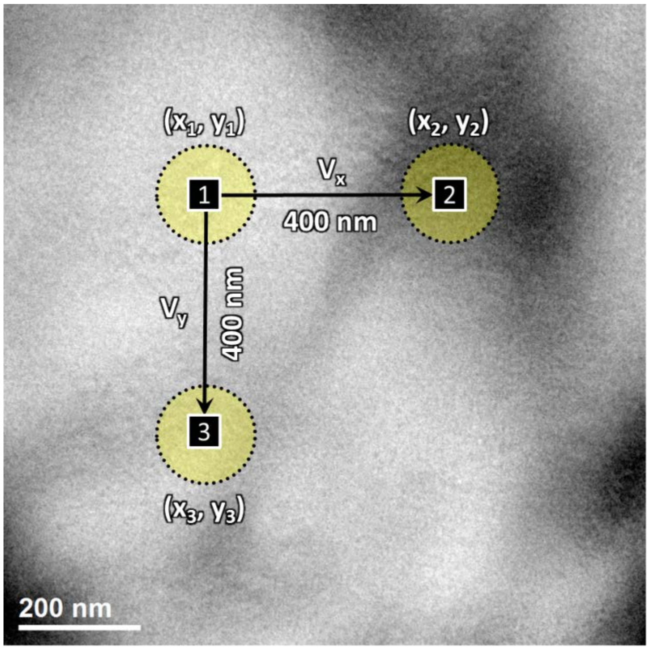


Figure 2.7: Schematic of calibration procedures for scanning electron diffraction

Figure 2.7 shows the calibration procedures for x and y directly obtained from EMGetBeamShift. The calibration procedure is carried out under a standard magnification that has been calibrated. First, the beam is positioned at '1' in Fig. 2.7. The reference value of (x_1, y_1) is then obtained using EMGetBeamShift from this initial beam position. Using the beam shift knob, the electron beam is horizontally shifted to '2' in Fig. 2.7, and the values of (x_2, y_2) are obtained. Using the calibrated magnification, the distance (d) between '1' and '2' can be set to a fixed value. Figure 2.7 shows an example for d=400 nm. From the above values, the horizontal scanning vector ($\mathbf{V_x}$) can be calculated as following:

$$\mathbf{V_x} = \frac{1}{d}(x_2 - x_1, y_2 - y_1)\,(\mathrm{nm}) \tag{2.8}$$

Similarly, the vertical scanning vector ($\mathbf{V_y}$) can be calculated from (x_1, y_1) and (x_3, y_3) as following:

$$\mathbf{V_y} = \frac{1}{d}(x_3 - x_1, y_3 - y_1)\,(\mathrm{nm}) \tag{2.9}$$

Once $\mathbf{V_x}$ and $\mathbf{V_y}$ are calibrated, the electron beam can be shifted to a position with specific distance using the combination of two vectors. The SED script can be found in Appendix A.

2.2.2 Symmetry quantification

Table 2.3: Image processing commands for symmetry quantification

Synopsis	Sum(Image)
Description	Calculate the sum of an image
Synopsis	Rotate(Image, Rotation angle)
Description	Rotates an image to the clock-wise by the rotation angle
Synopsis	FlipHorizontal(Image)
Description	Flips an image horizontally
Synopsis	FlipVertical(Image)
Description	Flips an image vertically

The symmetry quantification program mainly uses the commands given in Table 2.3. The symmetry recorded in CBED patterns can be quantified using either the *R*-factor or the cross-correlation coefficient [78, 79]. In this section, the script for the correlation coefficient will be described (see details in Ch. 4). The correlation coefficient is basically defined as

$$\gamma = \frac{\sum_{x,y}\left\{\left[I_A(x,y) - \overline{I_A}\right]\cdot\left[I_B(x,y) - \overline{I_B}\right]\right\}}{\sqrt{\left\{\sum_{x,y}\left[I_A(x,y) - \overline{I_A}\right]^2\right\}\cdot\left\{\sum_{x,y}\left[I_B(x,y) - \overline{I_B}\right]^2\right\}}} \tag{2.10}$$

, where $\bar{I}_A$ and $\bar{I}_B$ are the mean values of two images (I_A and I_B) that are being compared [79]. From the Eq. 2.10, the numerator and denominator have the exact same values if the two templates are absolutely identical. The correlation coefficient is close to 1 as the two images become identical.

As shown in Eq. 2.10, the correlation coefficient requires a pixel-by-pixel operation. For the pixel-by-pixel operation, `for` or `while` statements can be used for looping over the image pixels. This implementation drastically increases the computation time as the image size increases because access to individual image element takes extra operation steps in the script. In order to reduce the computation time, image-to-image process is used in the script for symmetry quantification. The following shows an example for calculating the correlation coefficient using image operations.

```
Number CrossCorrelation(Image ImgA, Image ImgB)
{
  Number Numerator, Denominator
  Numerator=sum((ImgA-mean(ImgA))*(ImgB-mean(ImgB)))
  Denominator=sqrt(sum((ImgA-Mean(ImgA))**2)*sum((ImgB-mean(ImgB)) **2))
  return Numerator/Denominator
}
```

In the above script, the correlation coefficient between two images is calculated without using the loop statements such as `for` and `while`. The image-based operation leads to a drastic decrease in the computation time. The full script source for the symmetry quantification can be found in Appendix B.

2.3 Operation issues for scanning CBED

2.3.1 Illumination system alignment

Ideally in performing scanning CBED, the electron probe is scanned at different sample position without change in beam tilt. The electron probe is positioned using the illumination deflection coils (Fig. 2.6). Improper alignment, however, causes beam tilt, which shifts the diffraction pattern during scanning. Figure 2.8 shows the direct beam movement recorded in the JEOL2010F. The electron beam was scanned over a Si single crystal for 20x1 data points with the step length of 10 nm. As shown in Fig. 2.8(b), the initial (x, y) position (11, 9) continually fluctuates during scanning.

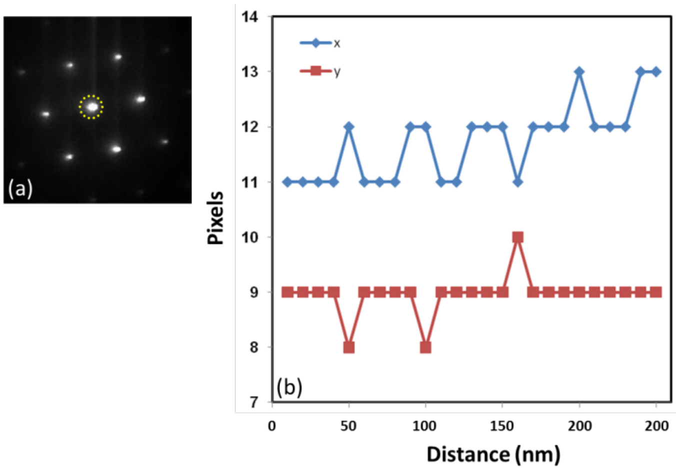


Figure 2.8: Direct beam movement during scanning with the misaligned TEM

The above issue originates mainly from the intermediate lens setting under diffraction mode and misalignment of the illumination system. Figure 2.9 shows the electron ray diagram under diffraction mode for (a) the strong and (b) the proper intermediate lens setting. The '1' and '2' represent the electron beam images coming from the different positions, and the electron ray diagram is only drawn for transmitted electrons. For Fig. 2.9(a), the intermediate lens is too strong so that the direct beam is shifted by the position of incident electron beam. In contrast, the direct beam does not move under the proper setting for intermediate lens as shown in Fig. 2.9(b).

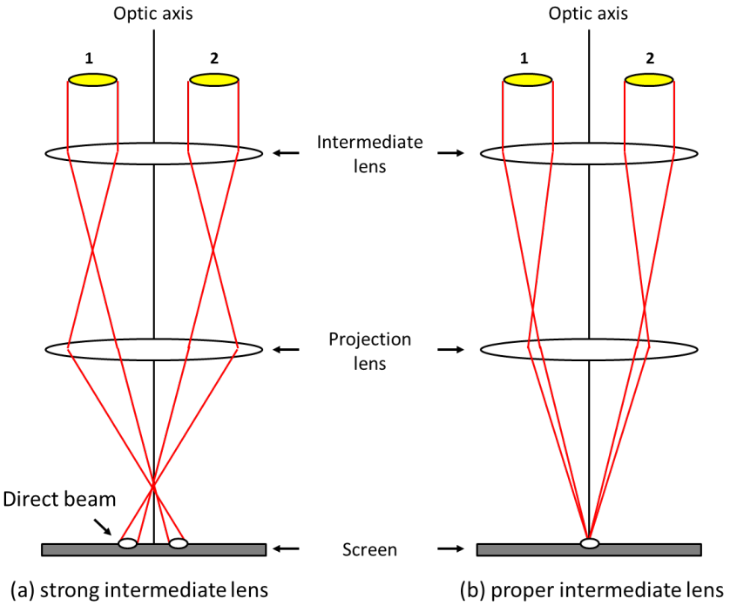


Figure 2.9: Effect of intermediate lens setting for the pattern movement under diffraction mode

The pattern movement can be minimized by changing diffraction focus[2]. The procedures for doing this are following:

(1) Align TEM following the standard alignment procedure for CL aperture, beam tilt, gun tilt, pivot point, etc.

(2) Find a vacuum area and focus the electron beam in TEM mode (MAG mode for JEOL TEM).

(3) Switch to diffraction mode. The direct beam will be seen on the screen. In this step, the direct beam is not necessarily sharp.

(4) Toggle (or press) BEAM SHIFT switch (or button) to the left position X and set the frequency to the maximum value.

---

[2] The alignment procedures are explained based on the JEOL TEM instruments. The JEOL TEMs have the diffraction focus knob to adjust the strength of intermediate lens under diffraction mode. Other TEM instruments (FEI, ZEISS, etc.) may have a different function to adjust the intermediate lens setting.

(5) Adjust the diffraction focus until the direct beam does not move.

(6) Toggle (or press) BEAM SHIFT switch (or button) to the left position Y and set the frequency to the maximum value.

(7) Adjust the diffraction focus until the direct beam does not move.

(8) Repeat the above procedures until the pattern movement is minimized.

(9) Once finish the alignment, do not change the diffraction focus.

The alignment for the illumination system includes adjustment for shift-tilt purity, which separates beam shift from beam tilt in the beam deflection coil and bright tilt for the alignment of optical axis of the objective lens. The mini lens plays an important role in the overall alignment. However, there is no independent way to align this lens.

2.3.2 Calibration for SCBED

To test above alignment procedures, we did a calibration for JEOL Cryo-2100 LaB_6 TEM and JEOL 2010F-FEG TEM. The scanning ED patterns were acquired from the Si single crystal. From the single diffraction pattern, a BF and ADF image was obtained by integrating the intensity of direct beam and the intensity of diffraction spots inside two circles (Fig. 2.10(a)), respectively. The calculation was repeated over all diffraction patterns to obtain the BF and ADF images. For the Si single crystal, the BF and the ADF image must display uniform contrast for the scanning area. As expected, the calculated BF (Fig. 2.10(a)) and ADF (Fig. 2.11(b) image show the uniform contrast.

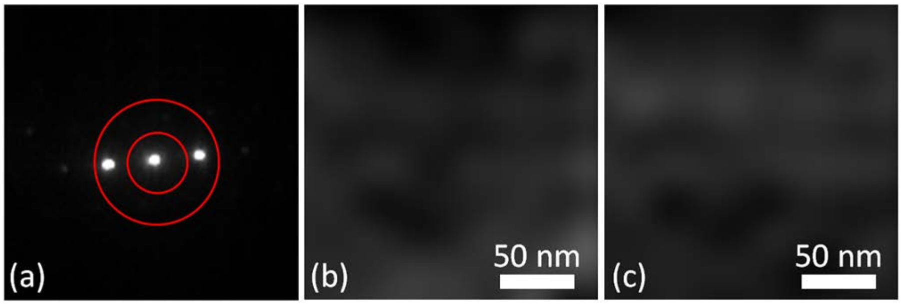


Figure 2.10: (a) The single electron diffraction pattern obtained from the Si single crystal. The scanning ED was carried out, and (b) and (c) are the calculated (a) BF and (b) ADF image from the acquired scanning ED patterns.

For comparison, the same experiment was performed on JEOL 2010F-FEG TEM. As shown in Figs. 2.11(b) and (c), however, the BF (Fig. 2.11(b)) and the ADF (Fig. 2.11(c)) image display the variations in contrast. These results show that the different lens system of TEM induces the beam tilting during scanning even though the instrument is properly aligned. In the following chapters, therefore, JEOL Cryo-2100 LaB_6 TEM is mainly used for the SCBED technique in order to avoid the artifact from beam tilting.

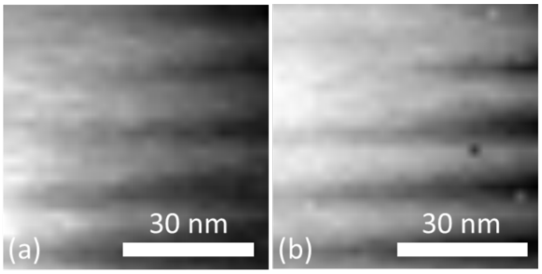


Figure 2.11: The calculated (a) BF and (b) ADF images show variations in contrast.

2.4 Convergent beam electron diffraction (CBED)

Convergent beam electron diffraction (CBED) is a powerful technique for the study of symmetry. It provides much more information than what can be obtained from XRD or conventional electron diffraction patterns (spot patterns as obtained by selected area diffraction) [80-82]. For comparison, the CBED pattern can be obtained by creating a focused beam with a convergence semiangle on the sample. X-ray diffraction (XRD) technique is widely used to determine the crystal symmetry. However, the obtained XRD information only indicates one of the 11 Laue groups and the possible presence of screw axes and glide planes [83, 84]. In comparison, the recorded CBED patterns can be used to determine various crystal symmetries including three dimensional crystallographic information. By carefully examining the symmetry of multiple CBED patterns for a given crystal structure, it is possible to determine both the point group and the space group of the crystal [69, 71, 82, 85].

The principle for measuring local crystal symmetry using CBED is illustrated in Fig. 2.7. A CBED pattern is recorded with a convergent electron beam that forms a focused probe on a thin crystal. The convergence angle α determines the diameter of CBED disk. The symmetry information is recorded in the diffraction disks as a specific pattern (also known as rocking curve information). The rocking curve information reflects the symmetry of the crystal. For example, the sample in Fig. 2.12 has two mirror planes (I and II). In the pattern symmetry, the rocking curve is symmetrical along the directions of two mirror planes.

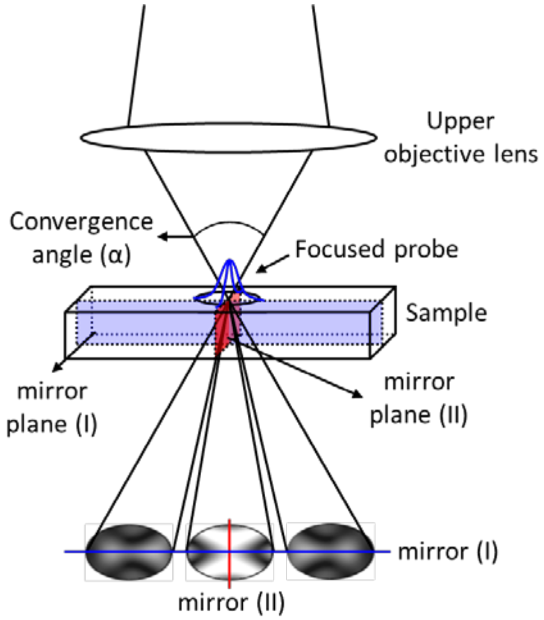


Figure 2.12: Schematic ray diagram for CBED pattern from a thin crystal with mirror symmetry

The CBED pattern symmetry is used to determine the crystal point groups. The symmetry determination starts with an investigation of the zero-order Laue zone (ZOLZ) symmetry details recorded in a zone axis pattern (ZAP) [85]. The symmetry in ZOLZ CBED pattern is classified into 10 two-dimensional (2D) point groups as shown in Table 2.3. The determination of 2D point groups in ZOLZ CBED pattern is based on the repeating patterns associated with the symmetry elements of a mirror or a rotation operation. Next, the obtained 2D point group is used to specify the projection diffraction groups and possible diffraction groups, which, in turn, determine the point group [85, 86]. A graphical representation of each diffraction group is provided with table showing how the 2D point groups are related to 32 point groups and 230 space groups [85]. Table 2.4 shows the relation between the diffraction groups and the crystal point groups.

Table 2.4: Relation between the observed pattern symmetry (ZOLZ) and the 31diffraction groups which correspond to the 32 different 3D point groups [86]

ZOLZ symmetry	Projection diffraction group	Possible diffraction groups	Symmetries of high-order information	
			Whole pattern	Zero-order (or bright field) disk
1	1_R	1	1	1
		1_R	1	2
		2	2	2
2	21_R	2_R	1	1
		21_R	2	2
m	$m1_R$	mR	1	m
		m	m	m
		$m1_R$	m	2m
2mm	$2mm1_R$	$2m_Rm_R$	2	2mm
		2mm	2mm	2mm
		2_Rmm_R	m	m
		$2mm1_R$	2mm	2mm
4	41_R	4	4	4
		4_R	2	4
		41_R	4	4
4mm	$4mm1_R$	$4m_Rm_R$	4	4mm
		4mm	4mm	4mm
		4_Rmm_R	2mm	4mm
		$4mm1_R$	4mm	4mm
3	31_R	3	3	3
		31_R	3	6
3m	$3m1_R$	$3m_R$	3	3m
		3m	3m	3m
		$3m1_R$	3m	6mm
6	61_R	6	6	6
		6_R	3	3
		61_R	6	6
6mm	$6mm1_R$	$6m_Rm_R$	6	6mm
		6mm	6mm	6mm
		6_Rmm_R	3m	3m
		$6mm1_R$	6mm	6mm

('R' is used to indicate the operation that rotates each dark-field disk by π about its own center)

Table 2.5: Relation between the diffraction groups and the crystal point groups [85]

Diffraction Group / Point Group	1	$\bar{1}$	2	m	2/m	222	mm2	mmm	4	$\bar{4}$	4/m	422	4mm	$\bar{4}2m$	4/mmm	3	$\bar{3}$	32	3n	$\bar{3}m$	6	$\bar{6}$	6/m	622	6mm	6m2	6/mmm	23	m3	432	$\bar{4}3m$	m3m
$6mm1_R$																											x					
$3m1_R$																										x						
6mm																									x							
$6m_Rm_R$																								x								
61_R																							x									
31_R																						x										
6																					x											
6_Rmm_R																				x												x
3m																			x												x	
$3m_R$																		x												x		
6_R																	x												x			
3																x												x				
$4mm1_R$															x																	x
4_Rmm_R														x																	x	
4mm													x																			
$4m_Rm_R$												x																		x		
41_R											x																					
4_R										x																						
4									x																							
$2mm1_R$								x							x												x		x			x
2_Rmm_R					x			x			x				x					x			x				x		x			x
2mm							x																			x						
$2m_Rm_R$						x						x		x										x				x		x		
$m1_R$							x						x	x											x	x					x	
m				x			x						x	x					x			x			x	x					x	
m_R			x			x	x		x	x		x	x	x				x			x			x	x	x		x		x	x	
21_R					x																											
2_R		x			x			x			x				x		x			x			x				x		x			x
2			x															x														
1_R				x															x													
1	x		x	x		x	x		x	x		x	x	x		x		x	x		x	x		x	x	x		x		x	x	

2.4.1 CBED simulation by the Bloch wave method

Experimental CBED patterns can be compared with simulated patterns. Such comparison greatly helps the crystal symmetry determination. The Bloch wave method is often used for CBED pattern simulation. Bloch wave (or Bloch state), named after Felix Bloch, refers the electron wave function satisfying the requirement of a periodic potential [87-89]. Bloch's theorem states that the wavefunction (or energy eigenstate) (ϕ), in a periodic crystal potential, can be written as the product of a plane wave and a function with the periodicity of the Bravais lattice:

$$\phi(\mathbf{r}) = C(\mathbf{r})\exp(2\pi i\mathbf{k}\cdot\mathbf{r}) \tag{2.11}$$

, where C(**r**) is the periodic function, and exp(2πi**k**·**r**) is the plane wave. Using Fourier series, C(**r**) can be expanded as following

$$C(\mathbf{r}) = \sum_g C_g \exp(2\pi i\mathbf{g}\cdot\mathbf{r}) \tag{2.12}$$

Thus, the *n*-th Bloch wave in the crystal is

$$\phi^{(n)}(\mathbf{r}) = \sum_g C_g^n \exp(2\pi i\mathbf{g}\cdot\mathbf{r})\cdot\exp(2\pi i\mathbf{k}^{(n)}\cdot\mathbf{r}) \tag{2.13}$$

The Bloch wave method, developed by Bethe in 1928, solves the dynamical diffraction of high energy electrons in a periodic crystal potential by expressing the electron wave function using Bloch waves [90]. Thus, electron wave function in the crystal can be the sum of Bloch waves:

$$\Psi(\mathbf{r}) = \sum_n c_n\phi^{(n)}(\mathbf{r}) \tag{2.14}$$

, where c_n is the excitation amplitude of n-th Bloch wave. The electron wave function must be a solution of the Schrödinger equation which describes the interaction of high-energy electrons with the crystal potential.

$$\frac{-h^2}{8\pi^2 m}\nabla^2\Psi(\vec{r}) - |e|V(\vec{r})\Psi(\vec{r}) = \frac{h^2 K_0^2}{2m}\Psi(\vec{r}) \tag{2.15}$$

,where h is Planck's constant, and K_0 (=1/λ) is the incident electron wave vector in vacuum. The static crystal Coulomb potential, V(**r**), can be expanded using Fourier series

$$V(\mathbf{r}) = \sum_g V_g \exp(2\pi i\mathbf{g}\cdot\mathbf{r}) \tag{2.16}$$

, where V_g is the Fourier coefficient of the crystal potential in volts, which is also known as the elctron scattering factor. We can now substitute the electron wave function (Ψ and $\nabla^2\Psi$) and the crystal potential for V(r) in Schrödinger equation (2.15). This gives

$$[\mathbf{K}^2 - (\mathbf{k}^{(j)} + \mathbf{g})^2]C_{\mathbf{g}}^{(j)} + \sum_{\mathbf{h}} U_{\mathbf{g}-\mathbf{h}}^c C_{\mathbf{h}}^{(j)} = 0 \tag{2.17}$$

$U_{\mathbf{g}}^c$ is the complex electron structure factor and defined as

$$U_{\mathbf{g}}^c = \frac{2m|e|V_g}{h^2} \tag{2.18}$$

An imaginary potential can be added to describe the depletion of the elastic wavefield by inelastic scattering (absorption). The total potential known as an optical potential is given by

$$U(\mathbf{r})=U^c(\mathbf{r})+iU^{'}(\mathbf{r}) \tag{2.19}$$

The Fourier components of the total potential are

$$U_g=U_g^c+iU_g^{'} \tag{2.20}$$

The ratio for $U_g^{'}/U_g^c$ is known as an absorption coefficient, which is real for centric crystals and complex for acentric crystals. The diffraction geometry is approximated by a parallel crystal slab with surface normal n. To satisfy the boundary condition

$$\mathbf{k}^{(\mathbf{j})}=\mathbf{K}+\gamma^{(\mathbf{j})}\mathbf{n} \tag{2.21}$$

for the *j*-th Bloch wave, where γ is the dispersion of wave vector inside the crystal. Equation 2.22 can be then simplified as following by considering only electron forward scattering (the backscattering term of γ^2 is neglected) [71]

$$2KS_g{}^2C_{\mathbf{g}}^{(j)}+\sum_{\mathbf{h}}U_{\mathbf{g}-\mathbf{h}}^cC_{\mathbf{h}}^{(j)}=2K_n\left(1+\frac{g_n}{K_n}\right)\gamma^{(j)}C_{\mathbf{g}}^{(j)} \tag{2.22}$$

Here, $K_n=\boldsymbol{K}\cdot\boldsymbol{n}$ and $g_n=\boldsymbol{g}\cdot\boldsymbol{n}$. Equation (2.22) is the basic equation of dynamical theory of transmission electron diffraction, including all high-order Laue-zone (HOLZ) effects, boundary inclination effects, and absorption terms. If the surface normal is opposite to the incident beam direction, g_n / K_n can be ignored due to $K_n \gg g_n$. Then, the equation can be simply written in matrix form

$$\mathbf{AC}^i = 2K_n \gamma^j \mathbf{C}^i \tag{2.23}$$

, where the off-diagonal entries of the structure matrix A are $U_{\mathbf{g-h}}$, while the diagonal entries are the excitation error terms $2KS_\mathrm{g}$. The structure matrix A is $n \times n$ for n-beams included.

$$\begin{pmatrix} \phi_0(t) \\ \phi_g(t) \\ \vdots \end{pmatrix} = \mathbf{C} \begin{pmatrix} \exp(2\pi i \gamma^1 t) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \exp(2\pi i \gamma^n t) \end{pmatrix} \mathbf{C}^{-1} \begin{pmatrix} \phi_0(0) \\ \phi_g(0) \\ \vdots \end{pmatrix} \tag{2.24}$$

The geometry of CBED is described by the zone axis coordinate (X, Y, Z) with origin at zone axis center and Z parallel to the zone axis (Fig. 2.13)

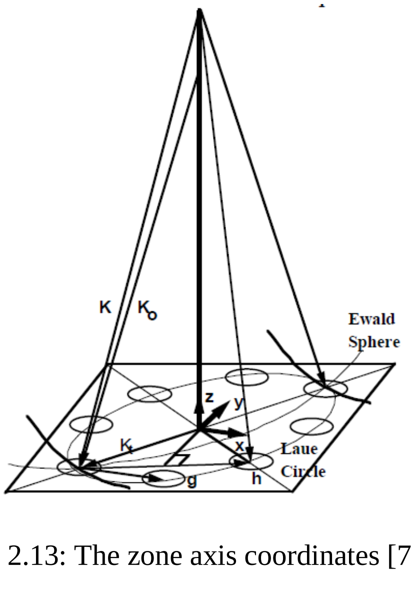


Figure 2.13: The zone axis coordinates [71]

Using the coordinate in Fig. 2.13, the diffraction geometry can be expressed in the simple mathematic form. The wave vector **K** of incident beam is specified by the tangential component of $\mathbf{K_t}$=x**X**+y**Y** and the diffracted beam at K_t+g_t. For each point inside the CBED disk of g, the intensity is then given by

$$I_g = \left|\phi_g(x,y)\right|^2 = \left|\sum_i c_i(x,y)C_g^i(x,y)\exp\left[2\pi i\gamma^i(x,y)t\right]\right|^2 \tag{2.25}$$

The diagonalization of structure matrix gives *n* eigenvalues ($K_n\gamma^j$) and *n* eigenvectors ($\mathbf{C}_g^i$). The amplitude of *n*-Bloch waves at crystal thickness *t* can be then obtained using *n* eigenvalues and eigenvectors calculated from (2.24). The term c_i is the first column of the inverse eigenvector matrix at *t*=0.

2.4.2 Electron and x-ray structure factor

The electron scattering factors are calculated using the Dolye-Turner atomic scattering factors for x-ray [91]. The Fourier coefficients of crystal potential with the Debye-Waller factor (B_i) is

$$V_g = \frac{1}{\Omega}\sum_i f^e(s)\exp(-B_i s^2)\exp(-2\pi i\mathbf{g}\cdot\mathbf{r}_i) \tag{2.26}$$

, where $s = \sin\theta_B / \lambda = \left|\mathbf{g}\right|/2$. The electron scattering factor $f^e(s)$ can be expressed in the Mott-Bethe formula

$$f^e(s) = \frac{\left|e\right|}{16\pi^2\varepsilon_0\left|s\right|^2}\left[Z - f^X(s)\right] \tag{2.27}$$

Here, $f^X(s)$ is the x-ray atomic scattering factor [92]. The electron scattering factors can be parameterized in a power series or an exponential expansion for computation [91, 93-95]. This study uses the Doyle-Turner parameterizations for the Bloch wave simulation [91]. The Doyle-Turner parameterization is exponential expansion:

$$f^e(s) = \frac{|e|}{16\pi^2\varepsilon_0}\left[\frac{Z - f^X(s)}{|s|^2}\right] = 0.04787801\sum_{j=1}^{4} a_j^i e^{-b_j^i s^2} \quad (\text{V}\cdot\text{nm}^3) \tag{2.28}$$

The units of Eq. 2.28 is V·nm^3 with s in nm^{-1}. Table 2.5 lists the parameters of a^i and b^i for Pb, Mg, Nb, Ti, O atoms, which are used in this study.

Table 2.6: Doyle-Turner parameters for Pb, Mg, Nb, Ti, and O atoms [91]

	Atomic No	**a^1**	**a^2**	**a^3**	**a^4**	**b^1**	**b^2**	**b^3**	**b^4**
Pb	82	4.785	3.688	1.5	0	0.27999	0.05083	0.00581	0
Mg	12	2.268	1.803	0.839	0.289	0.7367	0.20175	0.03013	0.00405
Nb	41	4.237	3.105	1.234	0	0.27415	0.05074	0.00593	0
Ti	22	3.565	2.818	1.893	0.483	0.81982	0.19049	0.0359	0.00386
O	8	0.455	0.917	0.472	0.138	0.2378	0.07622	0.02144	0.00296

The complex electron structure factor can be then calculated from V_g

$$U_{\mathbf{g}}^c = 0.006648352(1 + 1.956934 \times 10^{-6} E_0)V_g \quad (\text{Å}^{-2}) \tag{2.29}$$

, where the unit is Å^{-2} with V_g in volts.

CHAPTER 3

ION-MILLING INDUCED ARTIFICIAL DOMAIN STRUCTURES IN A PMN-PT SINGLE CRYSTAL[3]

This chapter reports an investigation of the domain structure in $Pb(Mg_{1/3}Nb_{2/3})O_3$-30%$PbTiO_3$ single crystals after ion milling. Ion milling induces microdomains of 100s nm in size. The induced microdomains disappear after temperature annealing or electric poling under applied electric fields leaving behind nanodomains of few nm in size. The microdomains are attributed to surface stress induced by ion milling. The results demonstrate the general importance of separating sample preparation artifacts from the true microstructure in the study of ferroic materials.

3.1 Introduction

Domains and their switching, under applied stresses and fields, are the most important characteristic of ferroic materials. In this communication, we report features of domain structures that are observed after ion-beam milling the piezoelectric ferroelectric single crystal PMN-PT, i.e., (1-x) $Pb(Mg_{1/3}Nb_{2/3})O_3$-x $PbTiO_3$, for x=0.30. Compositions in the PMN-PT system are finding increasing applications in new and improved electro-mechanical devices [13, 14].

Transmission electron microscopy (TEM) is routinely used for structural studies where specimens are thinned to electron transparency by mechanical polishing and ion-beam milling. Such preparation methods are known to introduce surface stresses and strains in metals and semiconductors [97-99]. Piezoelectrics by definition are responsive to mechanical forces, and the activity is particularly high for the new PMN-PT crystals. However, to the best of our knowledge,

---

[3] Reprinted with permission from K.-H. Kim, D. A. Payne and J.-M. Zuo, Appl. Phys. Lett. **97**, 261910. Copyright 2010, American Institute of Physics.

the effect of preparation conditions on domain structures observed in piezoelectrics by TEM has not been documented. Here, using commercial PMN-30%PT, i. e., x=0.3, we show that ion-beam milling induces domain features that are different from as received specimens. The results may be important for the next generation of high-frequency ultrasonic devices that are patterned by reactive-ion etching.

Compositions in the morphotropic phase boundary (MPB) region between rhombohedral and tetragonal states, particularly from the triple point (x=0.25) to the MPB (x=0.33), are known to have high electro-mechanical activity [55]. For the composition studied (x=0.30), the piezoelectric strain coefficient d_{333} =2000 ppm/V and the piezoelectric coupling factor = 0.94. For such compositions close to the MPB it has been proposed that an intermediary monoclinic phase exists, based upon X-ray and neutron diffraction studies, and polarized-light optical microscopy [24, 49, 51, 66, 67]. It has been claimed that the monoclinic phase is the key to the understanding of the outstanding properties, but significant doubt exists for macroscopic measurements on pulverized specimens [55]. Previous TEM studies of PMN-PT single crystals have revealed hierarchical nano- and micro-scale domain structures [64, 65, 100], which have been suggested to be a self-assembly of nano-domains of tetragonal structure leading to the overall formation of micro-domains with an averaged appearance of a monoclinic state.

3.2 Experimental methods

To study the hierarchical domain structure we selected a $[001]_C$ oriented PMN-30%PT slice from a melt-grown crystal. The specimen was mechanically thinned below 30 μm by polishing with a suspension of diamond particles (1 μm) before interleaving between two Cu grids without any chemical bond. The sandwich-like assembly was then Ar-ion milled at 5 kV/8 mA with a 15^o incident angle for less than 2 hours (Model 1010 Ion Mill, Fischione). The procedure for observing the domain structure in the TEM (JEOL 2010F FEG) was as follows. After the initial examination, the ion-milled specimen was annealed at 500^OC in air for 5 h to reduce any residual stress induced by mechanical polishing or ion milling [101]. Then, the specimen was observed a second time in the TEM to determine any difference in domain structure between the annealed specimen and the original ion-milled condition. The procedure was repeated several times through ion-milling and annealing. Since the thermal treatment would de-pole the specimen, the annealed specimens were re-poled at 4 kV/cm in air at a room temperature [101]. For consistency, the ion-milled specimens were also re-poled under identical conditions. The TEM results are as follows.

3.3 Results

Figure 3.1(a) shows typical medium magnification BF (Bright Field) image obtained from the ion-milled PMN-30%PT (This will be referred to as an as-milled specimen). The TEM images were recorded a few degree off the $[001]_C$ zone axis so as to obtain a clear contrast of domains. The domain structure shown in Fig. 3.1(a) was observed all over the sample. The first distinguishing feature of the as-milled PMN-30%PT is that the domain structure is comprised of micro-domains with widths ranging from the sub to tens of μm. The individual micro-domains consist of nano-domains with widths of 10 ~ 20 nm as shown (see the inset at the upper right of Fig. 3.1(a)). Compare with the domain structure of the annealed PMN-30%PT given in Fig. 3.1(b), which consists entirely of nano-domains, with widths of a few to several tens of nm (inset of Fig. 3.1(b)). That is, the micro-domains observed previously in the as-milled specimen disappeared after annealing. Therefore, to determine if the micro-domains observed in as-milled specimen were an artifact of ion milling, we ion milled the annealed sample for another 50 min. As shown in Fig. 3.1(c), the micro-domains originally observed in the as-milled specimen (Fig. 3.1(a)) reappear again. Also, the original nano-domains of 10 ~ 20 nm width are observed as well. The above results clearly demonstrate that the micro-domains observed in as-milled specimens and reported in earlier literatures are the result of ion milling [64, 65, 100].

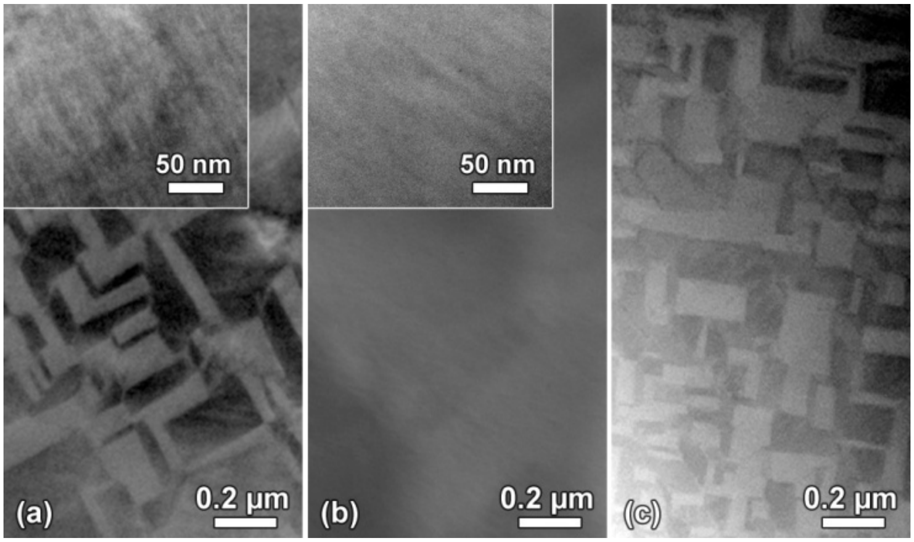


Figure 3.1: Bright field TEM images of ion-milled and annealed PMN-30%PT TEM samples. (a) and (c) the typical domain structure of PMN-30%PT after ion milling and (b) the microstructure after annealing.

To see if the as-milled and annealed specimens had different structures, convergent beam electron diffraction (CBED) was carried out on both specimens. CBED is very sensitive to crystal symmetry and the small focused probe is well suited for symmetry determination of nano-sized domains [82]. Figures 3.2 (a) and (b) show CBED patterns recorded from the as-milled and annealed specimens. Even though the overall domain structure of the two specimens is remarkably different, the symmetry of the CBED patterns from as-milled and annealed samples is similar. Both patterns approximately agree with the CBED pattern simulated at the zone axis of $[001]_T$ by using reported tetragonal crystallographic data for PMN-xPT [51]. Beside the overall agreement, there are significant deviations in the experimental CBED patterns from the tetragonal symmetry. However, our attempt to match the experimental CBED patterns by simulation using the reported structural data for other phases such as M_B (or M_A), M_C, and rhombohedral [51] has not been successful. The as-milled and annealed samples, therefore, are considered as pseudo-tetragonal structure.

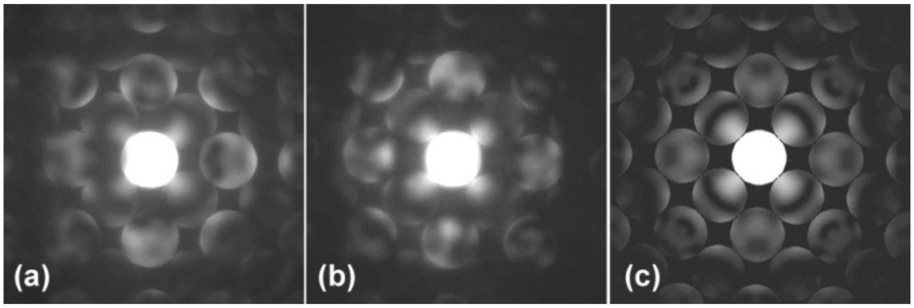


Figure 3.2: Convergent electron diffraction patterns recorded from (a) as-milled and (b) annealed samples at the zone axes of $[001]_C$ and c) simulated diffraction pattern at the zone axis of $[001]_T$ based on the tetragonal structure

Finally, the effect of poling is considered for as-milled and annealed specimens. Fig. 3.3(a) indicates that after poling the micro-domains observed previously in the as-milled condition mostly disappear in the annealed sample and very large domains are seen in the as-milled sample. Based upon this result, we believe the micro-domains observed after ion milling are not representative of a poled crystal, and the poled TEM specimen after annealing has the closest domain structure to the poled PMN-30%PT crystal before any TEM specimen preparation.

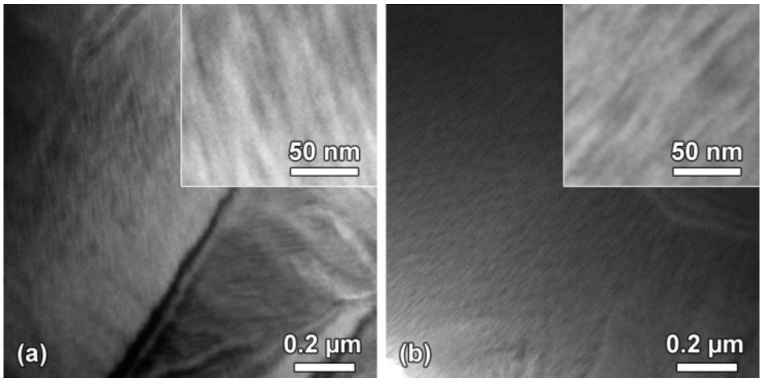


Figure 3.3: The domain structure of (a) the as-milled and (b) annealed PMN-30%PT TEM samples after poling

3.4 Discussions

A possible explanation for the ion-beam induced micro-domains follows. First, it is well known that the bombardment of Ar ions on TEM specimens can generate localized heating [102].

The PMN-PT phase diagram indicates a rhombohedral-tetragonal transformation at 90^{o}C, and a tetragonal-cubic transformation at 130^{o}C for PMN-30%PT [24]. The micro-domains could possibly result from if the ion miller heated the specimen and then quenched to a room temperature. However, the micro-domains shown in the as-milled condition were still observed when the specimen was ion milled with liquid nitrogen cooling (not shown here). Therefore, it can be concluded that the micro-domains in the as-milled specimens are not a result of phase transformations caused by temperature rise during ion milling.

Second, amorphization of TEM specimens during ion milling is considered next. In ion milling, the sample is not only thinned but also amorphized to some depth by ion bombardment [103]. The amorphous layer formed on the surface could induce a mismatch strain at the interface of the amorphous layer and the sample. This effect has been widely demonstrated in thin film fabrication [99, 104]. Bifano et al., for example, showed that ion-beam radiation of freestanding Si thin films yields compressive stress due to amorphization of the surface. The experimental conditions used by Bifano et al. are similar to our study [99]. An induced strain from ion milling, therefore, could lead to the development of a hierarchical domain structure in PMN-30%PT TEM specimens.

Figure 3.4 shows the electron diffraction patterns with Kikuchi lines recorded from two adjacent micro-domains in the as-milled sample with the same experimental conditions. The configurations of two electron diffraction patterns are exactly identical while Kikuchi lines cut through the different diffraction spots. Since the beam direction was fixed, this result shows that two adjacent micro-domains are miss-oriented by about 0.3^{o}. Thus, the PMN-30%PT crystal can accommodate induced strain by generating micro-domains that have a slight miss-orientation difference between each other. Such changes are known to occur even at small amounts of stress [105, 106].

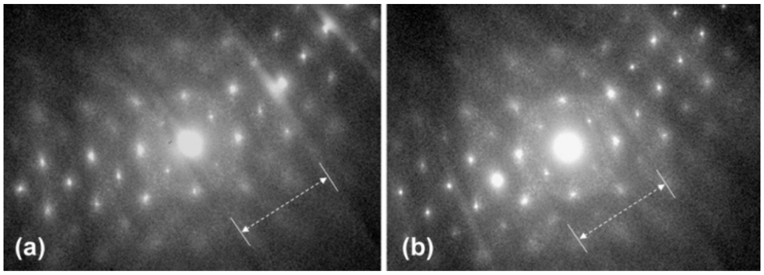


Figure 3.4: Electron diffraction patterns and Kikuchi lines were recorded from two adjacent microdomains in the as-milled sample. Kikuchi lines of Fig. 4(a) and (b) are tilted by 4.06^o and 3.78^o from the zone axis of $[001]_C$, respectively.

3.5 Summary and conclusion

In conclusion, we demonstrate that ion-beam milling induces a micro-domain structure in PMN-30%PT that is evidently different from the annealed or re-poled specimen. This artifact from milling is probably attributable to stresses that result from amorphorization of a surface layer. Both annealed and electric field-poled specimens maintain a relatively uniform nano-domain structure. The results may be applicable to ferroic materials in general, and of concern for the processing and patterning by ion–beam methods.

CHAPTER 4

SYMMETRY QUANTIFICATION AND SYMMETRY MAPPING[4]

We propose a new algorithm to quantify symmetry recorded in convergent beam electron diffraction (CBED) patterns and use it for symmetry mapping in materials applications. We evaluate the effectiveness of the profile *R*-factor (R_p) and the normalized cross-correlation coefficient (γ) for quantifying the amount of symmetry in a CBED pattern. The symmetry quantification procedures are automated and the algorithm is implemented as a DM (Digital Micrograph©) script. Experimental and simulated CBED patterns recorded from a Si single crystal are used to calibrate the proposed algorithm for the symmetry quantification. The proposed algorithm is then applied to a Si sample with defects to test the sensitivity of symmetry quantification to defects. Using the mirror symmetry as an example, we demonstrate that the normalized cross-correlation coefficient provides an effective and robust measurement of the symmetry recorded in experimental CBED patterns.

4.1 Introduction

Crystal symmetry often breaks down in real materials due to surface and interfacial stress and strain and the presence of defects. Thus, measurement of local symmetry can provide useful structural information that is difficult to obtain otherwise. While diffraction techniques, including convergent beam electron diffraction (CBED), are routinely used to determine the average crystal symmetry, determination of local crystal symmetry has received less attention. CBED using small electron probes is a powerful technique to extract symmetry information on the nanometer scale. Previous research has demonstrated that the zone axis pattern symmetry recorded in CBED

---

[4] Reprinted from Ultramicoroscopy, **124**, K.-H. Kim and J.-M. Zuo, *Symmetry quantification and mapping using convergent beam electron diffraction*, 71., Copyright (2013), with permission from Elsevier.

patterns can be used to determine all point groups without restriction [1-4]. In addition, the advance in electron microscope technology now makes it possible to form focused electron probes from 0.1 nm or smaller to tens of nanometers in diameter so it is possible to measure both the local symmetry at sub-unit cell level and the average symmetry with nanometer resolutions.

The symmetry recorded in the CBED patterns is in general determined by direct visual inspection [5], which does not provide an objective, uniform, measurement. Furthermore, experimental CBED patterns are often noisy and deviate from the ideal symmetry because of the sample geometry and defects. The imperfection in experimental CBED patterns can lead to uncertainty in the symmetry determination. To overcome this limitation, several groups have proposed alternative symmetry quantification procedures for CBED. An early method proposed by Mansfield was based on a comparison of geometrical features associated with HOLZ lines in recorded CBED pattern [6]. This method avoided direct comparison of diffraction intensities which were not readily available before the introduction of digital detectors for TEM [7]. The first use of intensity for symmetry quantification was described in the book by Tanaka et al., where they demonstrated the use of symmetry related line scans and the S-factor[8]. A potentially powerful technique was introduced by Hu et al. [9] for automated determination of symmetry in CBED patterns using the cross-correlation function and genetic algorithms. Despite these efforts, symmetry quantification has not been applied widely. There is a great need for robust, easy-to-use, method that can be applied directly to experimental CBED patterns. There is also a need to establish a common figure of merit so different CBED patterns can be compared directly.

Here, we propose a symmetry quantification method for CBED patterns. We compare the profile R-factor (R_p) [10, 11] and the normalized cross-correlation coefficient (γ) [12] for symmetry quantification. In order to measure the symmetry, regions in CBED patterns must be selected are aligned. We have developed computer algorithms to automate these procedures. We demonstrate that the method proposed here is highly effective and provides a more precise way to determine the symmetry in CBED patterns. The symmetry quantification method can be also combined with the scanning electron diffraction technique for symmetry mapping [13, 14].

4.2 Symmetry quantification methods

Figure 4.1 shows an experimental recorded CBED pattern from Si [110] zone axis. We use this pattern to demonstrate our image processing procedures for the dark-field symmetry quantification. The discussion below is specific to the mirror symmetry, but the principle also applies to rotational symmetry. First, two diffraction discs are selected about the mirror plane (yellow line) as shown in Fig. 4.1(a). For the calculation, the selected CBED discs are named as template A and template A' (Figs. 4.1(b) and (f)), respectively. Each template is then rotated by an angle θ so that the mirror is aligned as shown in Figs. 4.1(c) and (g). The template A is used as the reference motif so that the symmetry element is calculated by comparing with template A'. For the mirror operation, the template A' is flipped horizontally to obtain a mirror image as shown in Fig. 4.1(h). The mirror-applied image will be referred to as A'_m. For the rotational operation, the template A' is simply rotated by 180^o, 120^o, 90^o, and 60^o with respect to the 2-, 3-, 4-, 6-fold rotation, respectively. The rotated template A' will be referred to as A'_n (n=2, 3, 4, 6). The circular mask shown in Figs. 4.1(d) and (i) is used to remove areas affected by CBED disk edge. Thus, the final templates are obtained by multiplying the mask image to the templates A and A'_m as shown in Figs. 4.1(e) and (j).

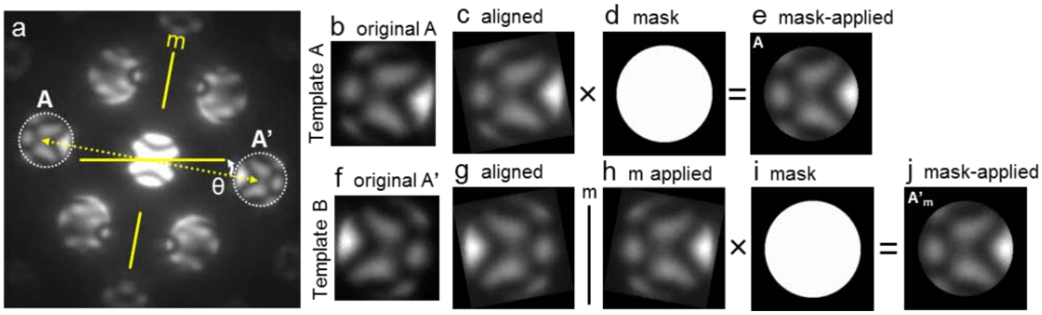


Figure 4.1: Image processing procedures used for symmetry quantification. The example here is for the mirror symmetry. Two diffraction discs related by mirror are selected as indicated by the dotted circles A and A' in the (a). Each disc is then processed to give two templates A and A' as shown above.

We used the profile *R*-factor (R_p) and the normalized cross-correlation coefficient (γ) to quantify the similarity between A and A'$_{\text{m or n}}$. The R_p and γ are calculated with the final templates of A and A'$_{\text{m or n}}$ based on pixel-by-pixel operation. Two templates have the same physical dimension with n by n pixels. The R_p is originally defined by the sum of squared intensity differences between an experimental profile and a computed profile scaled by the intensity sum of experimental profile [78, 108]. We replace the experimental and the computed profile with the templates A and B as the following,

$$R_p = \sqrt{\frac{\sum\left\{I_B\left(x,y\right)-I_A\left(x,y\right)\right\}^2}{\sum I_A(x,y)^2}} \tag{4.1}$$

, where $I_A(x, y)$ and $I_B(x, y)$ are an intensity of the image template A and B at (x, y), respectively. As two image templates are similar, the intensity difference sum between the template A and B approaches zero so that the smaller R_p value provides a better match.

The cross-correlation coefficient basically follows from the sum of multiplication of differences between the image template and a mean of the image template for the two templates.

The cross-correlation coefficient is then normalized as follows,

$$\gamma = \frac{\sum_{x,y}\left\{\left[I_A(x,y)-\overline{I_A}\right]\cdot\left[I_B(x,y)-\overline{I_B}\right]\right\}}{\sqrt{\left\{\sum_{x,y}\left[I_A(x,y)-\overline{I_A}\right]^2\right\}\cdot\left\{\sum_{x,y}\left[I_B(x,y)-\overline{I_B}\right]^2\right\}}} \tag{4.2}$$

, where $\overline{I}_A$ and $\overline{I}_B$ are the mean values of two templates [79]. From the Eq. 4.2, the numerator and denominator have the exact same values if the two templates are absolutely identical. In contrast to R_p, the cross-correlation coefficient is close to 1 as the two templates become identical.

The γ is widely used to quantify the similarity between two image objects. The average intensity is subtracted off in the calculation of γ, which makes it less sensitive to background intensity in diffraction patterns. The profile R-factor (R_p) is a commonly used goodness-of-fit (GOF) parameter, for example, it is used in Rietveld refinement to quantify the fit between an experimental and computed intensity profiles. The CBED pattern consists of recorded intensities over detector pixels so that the R_p can provide a quantitative measurement of the agreement between symmetry related diffraction discs. The R_p is sensitive to noise with the value increase with the amount of noise. The normalized cross-correlation coefficient (γ), on the other hand, is devised to quantify the correlation between two image templates. It has a maximum value of 1, which can advantageous in comparing diffraction patterns with different level of noises.

We implemented our symmetry quantification procedures in the scripting language of ‘DM (Digital Micrograph) script’, which is embedded in ‘Gatan Digital Micrograph Software’ (Gatan Inc.). The program developed for this study can quantify the symmetry elements of 2-, 3-, 4-, 6-fold rotation, and mirror in selected discs in a CBED pattern. The quantification result can be affected by the position of the template. The program automatically calculates the R_p and γ by moving the template B along the vertical and horizontal directions from the initial position. Then, the program repeats the diffraction pattern rotation, symmetry operation and template comparison procedures as described before. The R_p and γ values at each position are then stored, and the maximum correlation position is kept. The initial position of template B is then adjusted to the minimum R_P or the maximum γ position. Experimental CBED patterns with imperfect alignment induce a pattern shift inside CBED disks. The misalignment is taken into account in the calculation by reducing the mask size. The program uses a fractional number to define the mask size. For example, if the fraction is 1, the mask is created as same as the selected area. If the fraction is 0.5, the mask size is reduced by 1/2 from the selected area. Using, we are able to compensate small tilts away from the zone axis and obtain the maximum symmetry in the recorded patterns.

4.3 Symmetry quantification for the Si single crystal

We selected a Si single crystal with the zone axis of [110] to calibrate the proposed algorithm. The symmetry quantification was first performed with a simulated CBED pattern. The CBED simulation in this study uses a Bloch wave method [71] with the atomic scattering factors of Doyle and Turner [91] and the absorption parameters of Bird and King [109]. Figure 4.2 shows the simulated Si [110] CBED pattern at the thickness of 80 nm. A 2D point group of 2mm is quantified for the ZOLZ pattern, and the results are given in Table I. The 2-fold rotation symmetry is first investigated with the $(1\bar{1}\bar{1})$ / $(\bar{1}11)$ and $(1\bar{1}1)$ / $(\bar{1}1\bar{1})$ disc pairs. For the quantification, the $(1\bar{1}\bar{1})$ and $(1\bar{1}1)$ discs were used as references, and the $(\bar{1}11)$ and $(\bar{1}1\bar{1})$ discs were rotated by 180° so it can be compared with the reference. The corresponding R_p and γ values given in Table 4.1(a) are 0 and 1 for the $(1\bar{1}\bar{1})/(\bar{1}11)$ and $(1\bar{1}1)/(\bar{1}1\bar{1})$ pairs, respectively, which means the perfect matching between the symmetry related discs. For the mirror along $(2\bar{2}0)$ quantified in Table 4.1(b), three sets of $(1\bar{1}\bar{1})/(1\bar{1}1)$, $(00\bar{2})/(002)$ and $(\bar{1}1\bar{1})/(\bar{1}11)$ were selected. The $(1\bar{1}1)$, (002) and $(\bar{1}11)$ discs were flipped horizontally to make the mirror images of $(1\bar{1}1)_m$,

$(002)_{\mathrm{m}}$ and $(\bar{1}11)_{\mathrm{m}}$. The calculated R_p and γ values are 0 and 1, respectively. A second mirror along the (002) was also quantified using the same method, and the results are given in Table 4.1(c).

Table 4.1: Symmetry quantification results for the simulated Si [110] CBED pattern

Symmetry element	**Selected discs**	R_p	γ
(a) 2-fold	$(1\bar{1}\bar{1})$ / $(\bar{1}11)$	0	1
	$(1\bar{1}1)$ / $(\bar{1}1\bar{1})$	0	1
(b) m $(2\bar{2}0)$	$(1\bar{1}\bar{1})$ / $(1\bar{1}1)$	0	1
	$(00\bar{2})$ / (002)	0	1
	$(\bar{1}1\bar{1})$ / $(\bar{1}11)$	0	1
(c) m (002)	$(1\bar{1}\bar{1})$ / $(\bar{1}1\bar{1})$	0	1
	$(1\bar{1}1)$ / $(\bar{1}11)$	0	1

(The R_p is '0' to the third digit, and the γ is '1' for all practical purpose)

To demonstrate the sensitivity of our symmetry quantification procedure, we selected the $(\bar{1}1\bar{1})$ and $(1\bar{1}1)$ discs, which are related with the 2-fold rotation but not by mirror. The measurement for mirror symmetry show that the R_p drastically rises to 0.608, while the γ drops to 0.014. Thus, both R_p and γ are numerically capable of quantifying the amount of symmetry in the CBED pattern.

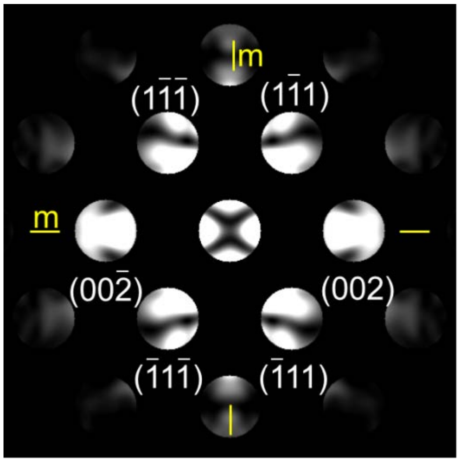


Figure 4.2: Simulated Si [110] CBED pattern. The two-dimensional symmetry of 2mm is observed at the pattern. The observed symmetry is quantified along two mirror positions and the 2-fold rotational axis.

Next, we applied the symmetry quantification to experimental patterns. Figure 4.3 shows an experimental Si [110] CBED pattern using JEOL 2100 LaB_6 TEM at 200 kV. For the $(2\bar{2}0)$ mirror, the R_p and γ values range from 0.041 ~ 0.096 and 0.981 ~ 0.991, respectively. Another mirror along the (002) has the R_p with 0.101 ~ 0.120 and the γ with 0.987 ~ 0.989. The 2-fold rotation symmetry for the $(1\bar{1}\bar{1})/(\bar{1}11)$ and $(1\bar{1}1)/(\bar{1}1\bar{1})$ pairs were quantified with the 2-fold rotation symmetry, and the results are given in Table 4.2(a). The R_p and γ values for the 2-fold rotation symmetry range from 0.059 ~ 0.122 and 0.983 ~ 0.989, respectively. In comparison to the simulation result, the R_p and γ deviate from the ideal value of 0 and 1.

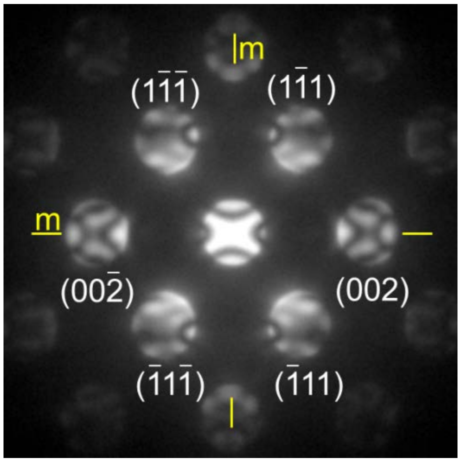


Figure 4.3: Experimental Si [110] CBED pattern. The observed symmetry of 2mm is quantified along two mirror positions and the 2-fold rotational axis.

Table 4.2. Symmetry quantification results for the experimental Si [110] CBED pattern

Symmetry element	**Selected discs**	R_p	γ
(a) 2-fold	$(1\bar{1}\bar{1}) / (\bar{1}11)$	0.059	0.983
	$(1\bar{1}1) / (\bar{1}1\bar{1})$	0.122	0.989
(b) m $(2\bar{2}0)$	$(1\bar{1}\bar{1}) / (1\bar{1}1)$	0.041	0.991
	$(00\bar{2}) / (002)$	0.056	0.981
	$(\bar{1}1\bar{1}) / (\bar{1}11)$	0.096	0.981
(c) m (002)	$(1\bar{1}\bar{1}) / (\bar{1}1\bar{1})$	0.101	0.987
	$(1\bar{1}1) / (\bar{1}11)$	0.120	0.989

To test the robustness of our symmetry quantification, we repeat the measurement over 20 experimental CBED patterns and plotted the results in Fig. 4.4 including the standard deviation obtained from the repeated measurements. In general, the γ values are more uniform than the R_p. The γ values are ranged from 0.981 to 0.991 for all quantification results. In contrast, the R_p shows variations from 0.036 to 0.115. The denominator of R_p consists of the intensity sum of the reference disc, while the numerator is simply the intensity difference between two discs. The R_p is therefore more sensitive to intensity changes than the γ, resulting in a large variation in the R_p. Based on the above results, the symmetry related experimental CBED discs in Si[110] are expected to have a 98% cross-correlation or better.

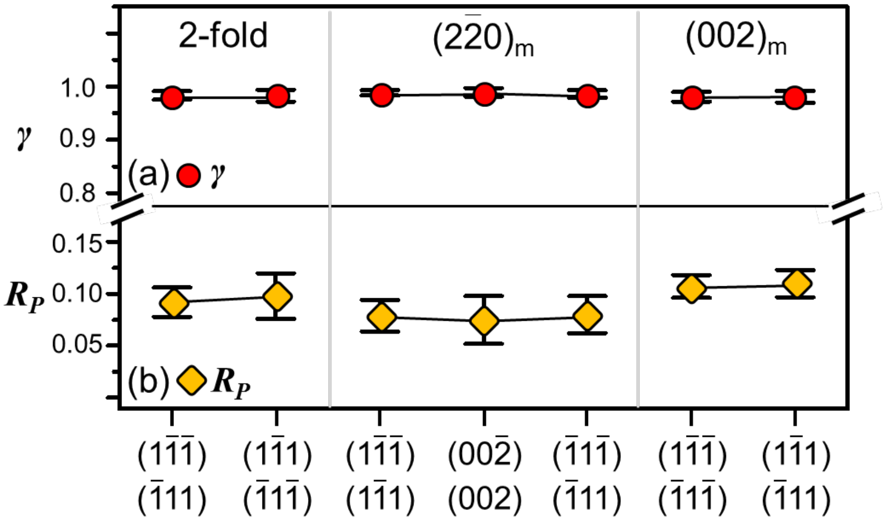
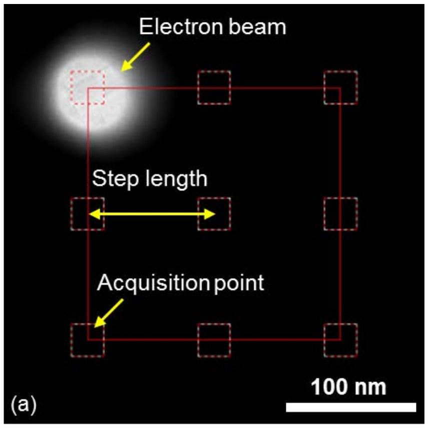


Figure 4.4: Averaged symmetry quantification results for 20 experimental [110] Si CBED patterns. The averaged γ and R_p for the selected discs are plotted in the (a) and (b), respectively.

4.4 Symmetry mapping with scanning electron diffraction technique

The proposed symmetry quantification method can be combined with the scanning electron diffraction technique [110, 111]. Figure 4.5 schematically shows the beam movement and data acquisition of the scanning electron diffraction program. The electron beam scans the sample over a defined rectangular grid. The distance between acquisition points is defined by a step length. Figure 4.5(b) illustrates the recording process for CBED patterns at each acquisition point. A series of CBED patterns are stored in a 4D dataset. The 4D dataset consists of m x n patterns; the m and n correspond to the number of sampling points along the two edges of the rectangular grid. The scanning CBED patterns tested here were recorded from 10 x 10 grids with a step length of 20 nm, which is indicated by the doted box in Fig. 4.6(a). The scanned area has a physical dimension of 180 x 180 nm^2. The symmetry is then mapped based on the proposed symmetry quantification method.



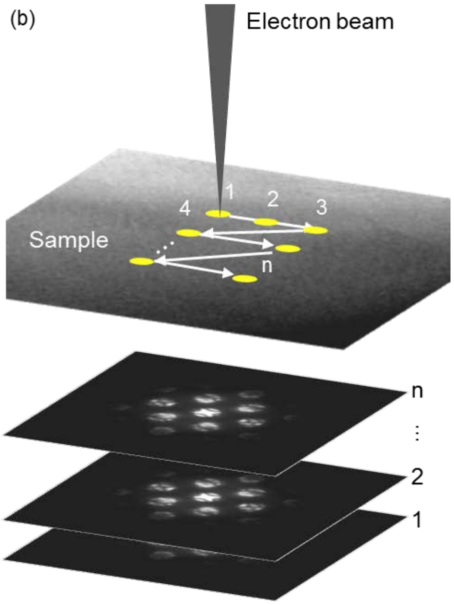


Figure 4.5: Schematic representation for the scanning electron diffraction (SED) program. The (a) shows the beam movement on the sample for 3x3 acquisition points with the step length of 100 nm. The (b) shows the recording process of CBED patterns during the scanning process.

The mirror along the white line in the inset is selected for the symmetry quantification. The discs of A, B and C are compared to the mirror images of A', B' and C', respectively. Figure 4.6(b) shows the symmetry map based on the R_p. The R_p symmetry map consists of very uniform values. The R_p has the lowest value with 0.08 and the highest value with 0.101. Figure 4.6(c) shows the symmetry map based on the γ. The γ symmetry map is also comprised of very uniform values. The γ values are ranged from 0.97 to 0.98. The R_p and γ symmetry maps are also represented as a surface plot. Figures 4.6(d) and (e) show the surface plots for the R_p and γ values. The level of surface in these plots is very flat as shown in the figures. The scan area, therefore, consists of very uniform symmetry with the mirror. The symmetry mapping based on the symmetry quantification and scanning electron diffraction technique can provide information about the symmetry fluctuation in the sample.

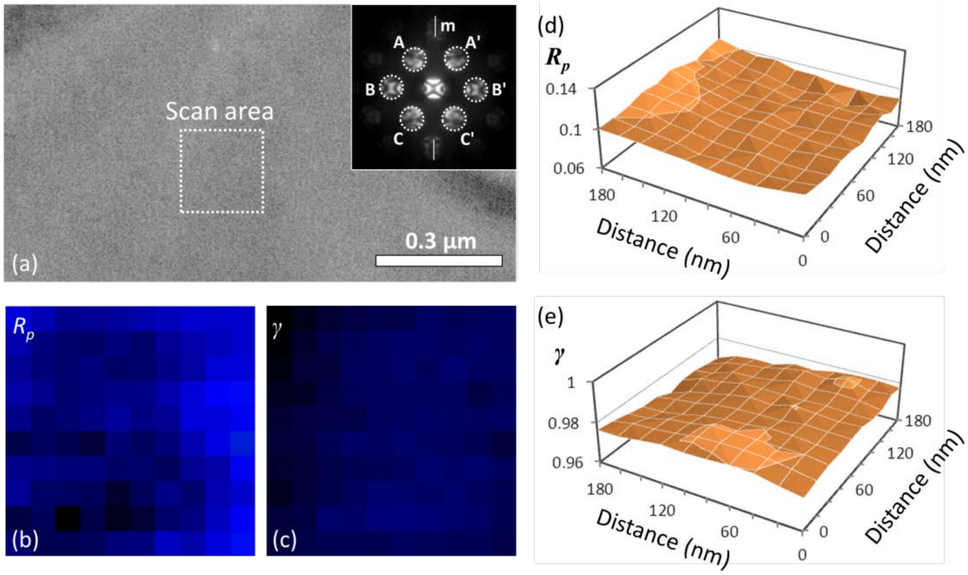


Figure 4.6: Symmetry quantification with the Si single crystal. The scanning CBED patterns were obtained from the dotted box in the (a), and the representative CBED patter is shown in the inset. The symmetry maps with (b) the R_p and (c) γ are calculated along the white line. The contrast for both of symmetry maps is very uniform with small variation. The (d) and (e) are the surface plots for the R_p and is γ. The surface plots of symmetry maps indicate the variation of symmetry in the Si single crystal is considerably small.

The symmetry mapping technique was then applied to a Si sample with defect in order to test the sensitivity of the developed algorithm to symmetry breaking caused by defects. A medium-magnification image presented in Fig. 4.7(a) shows a typical Bragg diffraction contrast for a stacking fault. The mirror selected for quantification is along the yellow line as indicated in Fig.

4.7(b) in the CBED pattern. The measurement used the A/A’, B/B’ and C/C’ disc pairs. From Fig. 7(a), the symmetry distribution was mapped on 20x10 grid points. A probe of 7.8 nm in FWHM (full-width half-maximum) was used for scanning CBED with a step length of 8 nm. Thus, the physical dimension of the scanned area is 152 x 72 nm^2. Figure 4.7(c) shows a magnified image of the investigated area, and Figs. 4.7(d) and (f) are the calculated symmetry maps for R_p and γ, respectively. The grid in the symmetry maps becomes bright as the symmetry of the investigated grid matches the selected symmetry (i.e., mirror). The symmetry map obtained with R_p is shown in reverse contrast for comparison. In both maps, the dark contrast indicates symmetry breaking from the selected mirror symmetry. For example, the profile of R_p and γ values were selected along the line indicated in Fig. 4.7(c) and plotted in Figs. (e) and (g), respectively. In the area of stacking fault, the R_p rapidly increases from 0.19 to 0.79, and the γ drops significantly from 0.98 to 0.18. Thus, the symmetry breaking is detected across the stacking fault and near the stacking fault. The two symmetry maps clearly show that the effect of stacking fault to the symmetry breaking is relatively localized because of the rigid shift of the stacking fault introduce little strain unless it is terminated by a partial dislocation.

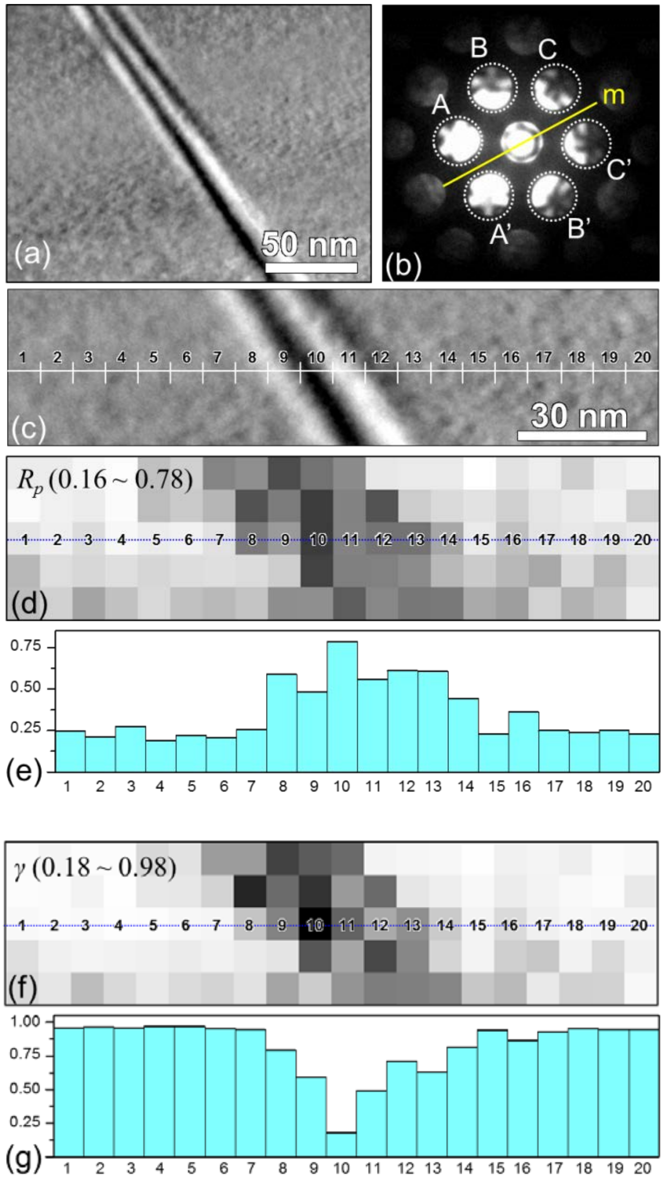


Figure 4.7: (a) A medium magnification image of strained silicon showing a stacking fault, (b) the selected CBED pattern from the investigated area, and (c) a magnified image of the area investigated by scanning CBED. The symmetry maps for R_p and γ are shown in Figs. 4.7(d) and (e), respectively. The (e) and (g) show the R_p and γ profile across the stacking fault along the line indicated in the (c).

4.5 Summary

We have proposed a symmetry quantification method by using the profile *R*-factor (R_p) and the normalized cross-correlation coefficient (γ). The Si single crystal is used to test the proposed symmetry quantification method. The result shows that the R_p of ~ 0.1 and the γ of ~ 0.98 can be obtained from a symmetrical experimental CBED pattern. The proposed symmetry quantification procedure is then combined with the scanning electron diffraction technique for symmetry mapping. The Si single crystal has a constant symmetry over the scanning area while the Si sample with defects shows large reduction in symmetry over a stacking fault. We believe that this study provides a powerful tool for studying symmetry in real materials.

CHAPTER 5

SYMMETRY OF (1-x)Pb($Mg_{1/3}Nb_{2/3}$)O_3-xPbTiO_3 (x=0.31) SINGLE CRYSTAL AT DIFFERENT LENGTH SCALES IN THE MORPHOTROPIC PHASE BOUNDARY REGION[5]

We use probes of three different length scales to examine symmetry of (1-x)Pb($Mg_{1/3}Nb_{2/3}$)O_3-xPbTiO_3 (PMN-xPT) single crystals in the morphotropic phase boundary (MPB) region at composition x=0.31 (PMN-31% PT). On the macroscopic scale, X-ray diffraction (XRD) shows a mixture of strong and weak diffraction peaks of different widths. The closest match to XRD peak data is made with monoclinic Pm (M_C) symmetry. On the local scale of a few nanometers, convergent beam electron diffraction (CBED) studies, with a 1.6 nm electron probe, reveal no obvious symmetry. These CBED experimental patterns can be approximately matched with simulations based on monoclinic symmetry, which suggests locally distorted monoclinic structure. A monoclinic Cm (M_A or M_B)-like symmetry could also be obtained from certain regions of the crystal by using a larger electron probe size of several tens of nm in diameter. Thus, the monoclinic symmetry of single crystal PMN-31%PT is developed only in parts of the crystal by averaging over locally distorted structure on the scale of few tens of nm. The macroscopic symmetry observed by XRD is a result of averaging from the local structure in PMN-31%PT single crystal. The lack of local symmetry at a few nm scale suggests that the polarization switching results from a change in local dipoles, which are not restricted to specific symmetry planes or directions.

---

[5] K.-H. Kim, David A. Payne and J.-M. Zuo, *Symmetry of piezoelectric (1−x)Pb($Mg_{1/3}Nb_{2/3}$)O_3-xPbTiO_3 (x=0.31) single crystal at different length scales in the morphotropic phase boundary region*, **86** (2012) 184113. Copyright (2012) by the American Physical Society.

5.1 Introduction

The symmetry of piezoelectric materials, such as (1-x)$Pb(Mg_{1/3}Nb_{2/3})O_3$-x$PbTiO_3$ and (1-x)$Pb(Zn_{1/3}Nb_{2/3})O_3$-x$PbTiO_3$ (known as PMN-xPT and PZN-xPT, respectively), has been widely studied for the simple reason that symmetry controls displacements of ionic charge and position, which, in turn, determines directions of spontaneous polarization ($\mathbf{P}_S$) and spontaneous strain ($\boldsymbol{\varepsilon}_S$), and field (E)-induced orientations of ferroelectric and piezoelectric properties. The high-temperature (HT) phase of PMN-xPT and PZN-xPT is cubic Pm3m with no spontaneous distortions. According to published x-T phase diagrams for PMN-xPT and PZN-xPT, the prototypic HT phase spontaneously distorts to rhombohedral (R) R3m symmetry on cooling at low x, or tetragonal (T) P4mm symmetry at higher x, in which $\mathbf{P}_S$ and $\boldsymbol{\varepsilon}_S$ are constrained to the cubic (noted by the subscript 'C') $[111]_C$ and $[001]_C$ directions for the R and T phases, respectively. The R and T phases are initially separated by a vertical boundary termed the morphotropic phase boundary (MPB), i.e., a chemically (x)-driven change in morphology.[25, 26] The MPB of PMN-xPT and PZN-xPT is defined in a narrow composition region where the R and T phases meet. This phase boundary composition has attracted much attention because displacements maximize as the lattice softens and transforms, giving rise to large enhancements in piezoelectric properties.[13, 14, 55]

A large body of work has been reported on the structural origin of the piezoelectric properties for PMN-PT at the MPB. A new phase with monoclinic (M) symmetry was proposed in the vicinity of MPB [24, 49, 51, 52, 66] as identified on pulverized powder samples using X-ray and neutron diffraction studies. [26, 46, 47] According to the notations proposed by Vanderbilt and Cohen, the monoclinic phase belongs to M_A or M_B (Cm) or M_C (Pm), in which $\mathbf{P}_S$ is aligned along $[uuv]_C$ (u>v) and $[0uv]_C$ (u<v) directions, respectively [48]. The M phase is said to be a structural bridge that facilitates polarization rotation from the R to T phases, which is atypical of ionic displacements. Several research groups, however, have disputed whether the observed M

phase truly has the monoclinic symmetry at the local (microscopic) scale [59, 113, 114]. The adaptive phase model proposed by Viehland and coworkers states that the M phase found in the MPB region is not a local symmetry but an averaged symmetry obtained from twin-related domain structures [58]. Wang et al. from transmission electron microscopy (TEM) studies suggests the monoclinic symmetry is a result of averaging over R or T nanodomains, which supports the adaptive phase model [64, 65, 100]. Another point-of-view, put forward by Kisi et al., suggests the M phase is not a true phase but a distorted structure resulting from residual stress; and the observed M phase is indeed neither sufficient nor necessary for an explanation of the large piezoelectric response in the MPB region [59]. Thus, a determination of symmetry, from the local to macroscopic level, in the MPB region, is critical to settle these disputes, and is the purpose of this investigation.

In this study, we selected a PMN-31%PT single crystal for symmetry determinations, which is an established MPB composition with reported properties [115]. Previously, the symmetry of PMN-xPT was investigated by optical microscopy, high resolution XRD, and neutron diffraction [24, 49, 51, 66], PMN-xPT, however, is known for having complex hierarchical domain structures, starting from nanodomains (of a few, to tens of nanometers,) to microdomains (of tenths, to tens of microns). The symmetry determination of nanodomains, therefore, requires a small diameter probe with nanometer resolution in order to determine the local symmetry. Considering this limitation, CBED performed in a TEM is an appropriate tool for a determination of local symmetry [71, 85, 116-118], The rocking curve information recorded in the CBED patterns is very sensitive to the symmetry of the crystal structure [81, 104, 119]. Local symmetry within a few nm can be studied with a field emission gun, which can provide electron probes of ~ 2 nm in diameter or less. The symmetry over several tens of nm can be investigated with a thermionic electron source, which forms a larger probe size of tens of nm. Previous symmetry studies were performed on powders or ion-milled specimens [24, 49, 51-53, 64, 65, 100]. However, the domain structure of PMN-xPT is very sensitive to specimen history, especially, applied and residual stresses, electric poling and heat-treatment conditions [96]. In this study, all specimens were annealed to attain the original structure in order to minimize the effects of stresses induced by polishing and ion-milling [96]. The symmetry of PMN-31%PT single crystal was then determined from CBED patterns obtained along several zone axes, and by comparison with electron diffraction simulations. In addition, a recently proposed algorithm was used to quantify the symmetry recorded in the experimental CBED patterns through use of a cross-correlation coefficient [107]. The physical insight will be useful for an understanding of the structure-property relations of the piezoelectric crystals.

5.2 Experimental

PMN-31%PT specimens were selected from a melt-grown crystal which had been sliced normal to the three principal orientations, $[001]_C$, $[010]_C$, $[011]_C$, and $[111]_C$,and thinned to less than 40 µm by mechanical polishing. Details of the specimen preparation are reported elsewhere [96]. An Ar-ion beam (4.5 kV, 6^o incidence) was used to mill the specimens for perforation and electron transparency. (Precision Ion Polishing System, PIPS™, Gatan, USA). It is known that ion-milling induces artificial domain structures in piezoelectric materials from surface stress, and so specimens were annealed at 500^oC in air with slow cooling. After annealing, XRD (PANalytical X’pert MRD system, Philips) was used to determine lattice parameters before TEM studies with two different beam sizes. Selected area electron diffraction (SAED or spot patterns) and CBED patterns were recorded along the zone axes of $[001]_C$, $[011]_C$ and $[111]_C$, respectively. We used a focused beam of 1.6 nm in FWHM (Full-Width Half-Maximum) in JEOL 2010F-FEG, and 35 nm in JEOL 2100 LaB_6, for the CBED studies. As mentioned, symmetry obtained with the smaller 1.6 nm beam will be referred to as local symmetry, whereas, symmetry obtained with the 35 nm probe will be termed the averaged symmetry obtained over multiple nanodomains. We also recorded SAED spot patterns which were indexed on reported crystallographic data.

For the symmetry determination, CBED patterns were recorded from different areas of the specimen in numbers ranging tens to hundreds in an effort to search for the highest symmetry. Theoretical CBED patterns were then simulated based on diffraction pattern indexing, and compared with the experimental CBED results, for a determination of crystal symmetry. The symmetry recorded in the CBED patterns was quantified through use of a cross-correlation coefficient (γ). Details for the symmetry quantification are explained in Ch. 5.3 [107].

5.3 Crystal structure and electron diffraction simulation

Several crystal structures have been reported for PMN-xPT depending on x and T [24, 120-122]. Slodczyk et al. reported a R structure with R3m symmetry for x= 0.09 and T= 12K [123]. The lattice parameters were a=b=c=4.0364 Å, and α=89.8826°, and with positional coordinates Pb (0, 0, 0), Ti/Nb/Mg (0.534, 0.534, 0.534), and O (0.541, 0.541, 0.03). These data were obtained by powder XRD. Here, we use rhombohedral axes, rather than hexagonal axes (used by others), for a direct comparison between orthogonal axes and structural models.

The reported M structures of PMN-xPT belong to two different space groups of monoclinic symmetry, according to the notation of Vanderbilt and Cohen [48], M_B (or M_A) belongs to space group Cm with lattice parameters a=5.6951 Å, b=5.6813 Å, c=4.0138 Å, and β =90.136°, and positional coordinates Pb (0, 0, 0), Ti/Nb/Mg (0.5250, 0, 0.498), O_1 (0.54, 0, -0.01), and O_2 (0.317, 0.267, 0.48). M_C belongs to space group of Pm with lattice parameters a=4.0183 Å, b=4.0046 Å, c=4.0276 Å and β =90.146°, and positional coordinates Pb (0, 0, 0), Ti/Nb/Mg (0.509, 0.50, 0.5479), O_1 (0.47, 0, 0.57), O_2 (0.417, 0.5, 0.059), and O_3 (-0.02, 0.5, 0.57). These data for Cm (M_A or M_B) and Pm (M_C) symmetry were obtained by powder XRD for x=0.29 and 0.32, respectively, at room temperature [51],

Singh et al. reported a T structure with space group P4mm for x= 0.39 at room temperature. The lattice parameters were a=3.9920 Å and c=4.0516 Å at x=0.39 positional coordinates Pb (0, 0, 0), Ti/Nb/Mg (0.5, 0.5, 0.532), O_1 (0.5, 0.5, 0.054), and O_2 (0.5, 0, 0.601).[9]

All the electron diffraction simulations here use the above reported crystal structures, the atomic scattering factors of Doyle and Turner [91], and the absorption parameters of Bird and King [91, 109]. The cation substitutions were treated as random, and the occupancy factors for Mg/Nb/Ti were calculated as 0.23, 0.46, and 0.31, respectively, for x = 0.31.

5.4 Results

5.4.1. X-ray diffraction from the $[001]_C$ oriented PMN-31%PT single crystal

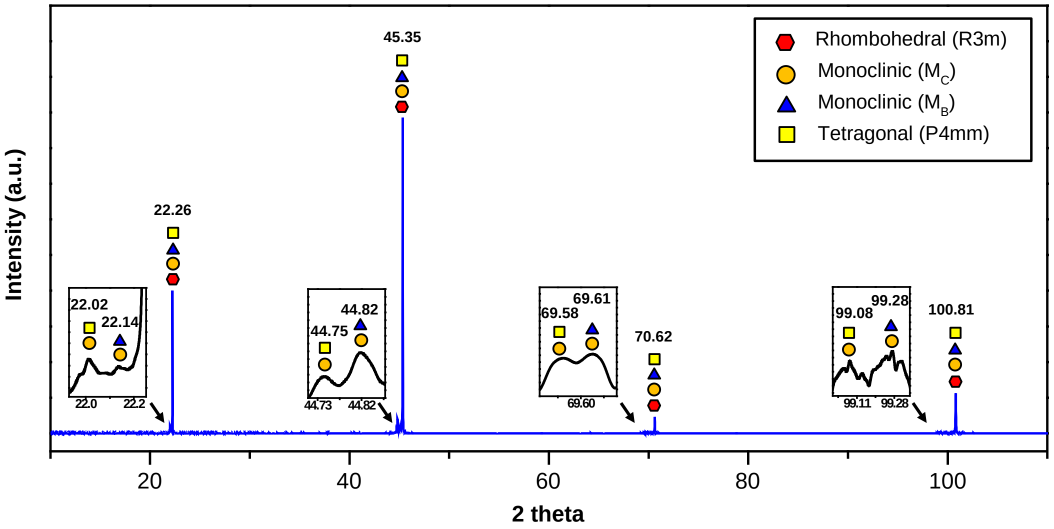


Figure 5.1: 2θ XRD data obtained from annealed PMN-31%PT. The same crystal was studied by electron diffraction. The diffraction peaks are labeled by numbers and indexed with symbols for different structures listed in the figure legend.

Figure 5.1 shows 2θ scan by X-ray obtained from an annealed crystal, which was later used in TEM studies. The diffraction pattern consists of strong diffraction peaks accompanied by a number of very weak diffraction peaks. The strong diffraction peaks and their sharpness indicate the quality of the annealed crystal. The high intensity peaks have 2θ values of 22.26^o, 45.35^o, 70.62^o, and 100.81^o. The weak peaks are clearly separated, as highlighted by the insets in Fig. 5.1, at 2θ values of ~ 22^o, ~ 44^o, ~ 69^o, and ~ 99^o. The FWHM of the strong diffraction peaks is about 0.02^o, which is close to the instrument resolution of 0.013^o in 2θ. The weak diffraction peaks are significantly broader (5x) than the strong diffraction peaks. Our 2θ values are compared with values calculated from the literature in Ch. 5.3 (Table 5.1). Calculations are for a variety of compositions with different x values (see above), but the compositions of the reported M_B (x=0.29) and M_C (x=0.32) are close to PMN-31%PT. Our single crystal XRD data was then indexed with all referenced structures and marked as shown in Fig. 5.1. Table 5.1 summarizes the indexing results. The M_C structure could be indexed on all the diffraction peaks in the experimental XRD data. Never-the-less, the average structure for PMN-31%PT cannot be determined unambiguously from the XRD data alone, and is a topic of our TEM studies that follow.

Table 5.1: Measured and calculated 2Θ angles and their differences. The calculation is based on the reported crystal structures of PMN-xPT [9-10, 34]. The composition (x) of the referenced crystal structure is specified in the table.

Exp. (x=0.31)	**R3m (x=0.09)**			**M_C (x=0.29)**			**M_B (x=0.32)**			**P4mm (x=0.39)**		
2Θ	**(h, k, l)**	**2Θ**	**$\Delta 2\theta$ (deg)**	**(h, k, l)**	**2Θ**	**$\Delta 2\theta$ (deg)**	**(h, k, l)**	**2Θ**	**$\Delta 2\theta$ (deg)**	**(h, k, l)**	**2Θ**	**$\Delta 2\theta$ (deg)**
22.02				001	22.04	0.02	-	-	-	001	21.92	-0.1
~ 22.14	-	-	-	100	22.10	-0.04	110	22.07	-0.04	-	-	-
22.26*	001	22.08	-0.18	010	22.17	-0.09	001	22.12	-0.14	100	22.26	0
44.75	-	-	-	002	44.96	0.21	-	-	-	002	44.70	-0.05
44.82				200	45.07	0.25	220	45.02	0.1	-	-	-
45.35*	002	45.04	-0.31	020	45.26	-0.09	002	45.12	-0.23	200	45.41	0.06
69.58	-	-	-	003	70.00	0.42	-	-	-	003	69.56	-0.02
69.61	-	-	-	300	70.18	0.57	330	70.14	0.53	-	-	-
70.62*	003	70.12	-0.5	030	70.46	-0.16	003	70.27	-0.35	300	70.75	0.13
~ 99.08	-	-	-	004	99.77	0.69	-	-	-	004	99.02	-0.06
~ 99.28	-	-	-	004	99.77	0.49	440	100.01	0.73	-	-	-
100.81*	004	99.98	-0.83	040	100.55	-0.26	004	100.23	-0.58	400	101.04	0.33

('*' indicates strong peaks observed in the experimental XRD)

5.4.2. Symmetry determination at several nanometers scale

Our symmetry determinations were carried out on annealed specimens, which are closest to the structure of unpoled PMN-31%PT single crystal. A typical bright field (BF) image of annealed PMN-31%PT is given in Fig. 5.2(a). This image has similar contrast and is comprised of fine nanodomains as illustrated in the inset of Fig. 5.2(a). A BF image of the ion-milled sample is given in Fig. 5.2(b). By comparison, the ion-milled sample consists of microdomains with widths of a few tenths of microns. Never-the-less, the individual microdomains in Fig. 5.2(b) are also comprised of nanodomains, as reported previously [96].

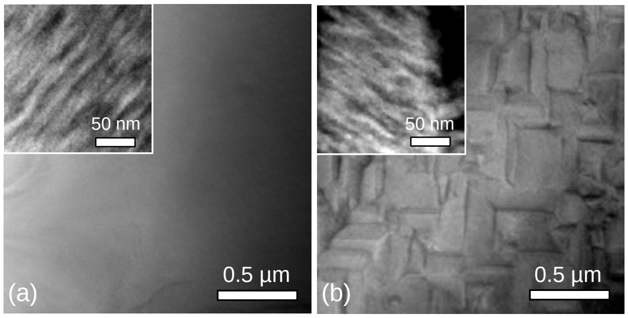


Figure 5.2: Bright-field images recorded from (a) the annealed PMN-31%PT crystal and (b) after ion milling. The image in (a) shows uniform contrast at medium magnification while the image in (b) shows submicron domain structure. The insets in (a) and (b) are for high magnification showing nandomains of ~10 nm in width in PMN-31%PT.

For the selection of the optimum combinations of zone axis patterns (ZAPs) necessary for symmetry determinations, we compared the space group, point group, and corresponding zone-axis symmetry of the diffraction group for each proposed structural model of PMN-xPT [85], as listed in Table 5.2. Each zone axis has the distinguished symmetry in the ZOLZ CBED pattern. To make use of Table 5.2, we first investigated the sample orientations of $[001]_C$ and $[010]_C$. The symmetry of PMN-31%PT single crystals will be determined by following ZAPs from $[001]_C$ to $[111]_C$ with the help of CBED simulations.

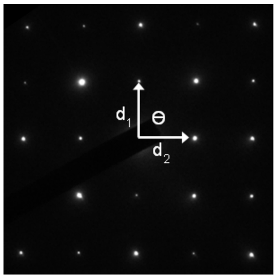
The focused beam probe of 1.6 nm in FWHM was used to investigate the local symmetry. What follows is an account of comparison between experimental CBED patterns recorded using this small probe and simulations. We show that experimental patterns contain no obvious symmetry and they can be approximately matched with simulations based on the monoclinic symmetry.

Table 5.2: Observable symmetry in CBED in the zero-order Laue zone (ZOLZ) for the reported crystal structures of PMN-PT and their space group (SG) and point group (PG)

Crystal Structure	SG	PG	Symmetry in the zero-order Laue zone						
			[001]	[010]	[100]	[011]	[101]	[110]	[111]
R	R3m	3m	m	m	m	m	m	1	3m
T	P4mm	4mm	4mm	m	m	m	m	m	m
M_B	Cm	m	m	1	m	1	m	1	1
M_C	Pm	m	m	1	m	1	m	1	1

5.4.2.1 Local symmetry along $[001]_C$ and $[010]_C$

Figure 5.3 shows the spot diffraction pattern obtained along zone axis $[001]_C$. The recorded diffraction pattern was indexed with the reported structures of PMN-xPT. The spot patterns consist of sharp and single crystal diffraction peaks without any peak splitting or additional diffraction peaks. According to the reported phase diagram of PMN-xPT, x=0.31 lies within the MPB region [24]. Diffraction pattern indexing was carried out on all possible crystal structures considering the experimental error. For example, the spot diffraction pattern of Fig. 5.3 can be indexed as the zone axes of $[001]_T / [010]_T$, $[001]_{M_C} / [010]_{M_C} / [100]_{M_C}$, $[001]_{M_B} / [110]_{M_B}$ and $[001]_R / [010]_R / [100]_R$. These indexing results are considered for the CBED simulations.



d	nm^{-1}	Error
d_1	2.421	± 0.016 nm^{-1}
d_2	2.439	
d_2/d_1	1.007	-1.3 ~ 1.4% (0.994 ~ 1.021)
Θ	~ 90°	

Figure 5.3: Selected area electron diffraction pattern recorded along $[001]_C$

Figures 5.4(a) and (b) show two examples of experimental CBED patterns recorded from $[001]_C$ and $[010]_C$ zone axes. We quantified the amount of mirror symmetry in the recorded CBED patterns in 4 directions that are marked as I, II, III, and IV in Fig. 5.4(a). Quantification was based on cross correlations between pairs of diffraction discs expected to have mirror symmetry.[107] For example, a mirror along I was quantified for Figs. 5.4(a) and (b) using the disc pairs of 1/7, 2/6, and 3/5 (for labeling, see Fig. 5.4(a)). Similar procedures were carried out for possible mirror symmetry along II, III or IV. The highest cross-correlation coefficient (γ) was detected along direction I, i.e., $\gamma_{m(I)}$=42% (the subscript 'm' with the parentheses '()' indicates the mirror quantification along the direction in parentheses) for the $[100]_C$ pattern. The $[010]_C$ pattern gave $\gamma_{m(II)}$=57% along direction II. For reference, a perfect mirror symmetry quantified for single crystal silicon gave γ_m values between 98 and 99% with a variation of ~ 3%.[107] Based on the low cross-correlation coefficients, the experimental CBED patterns show trivial, 1-fold rotation symmetry, along $[001]_C$ and $[010]_C$.

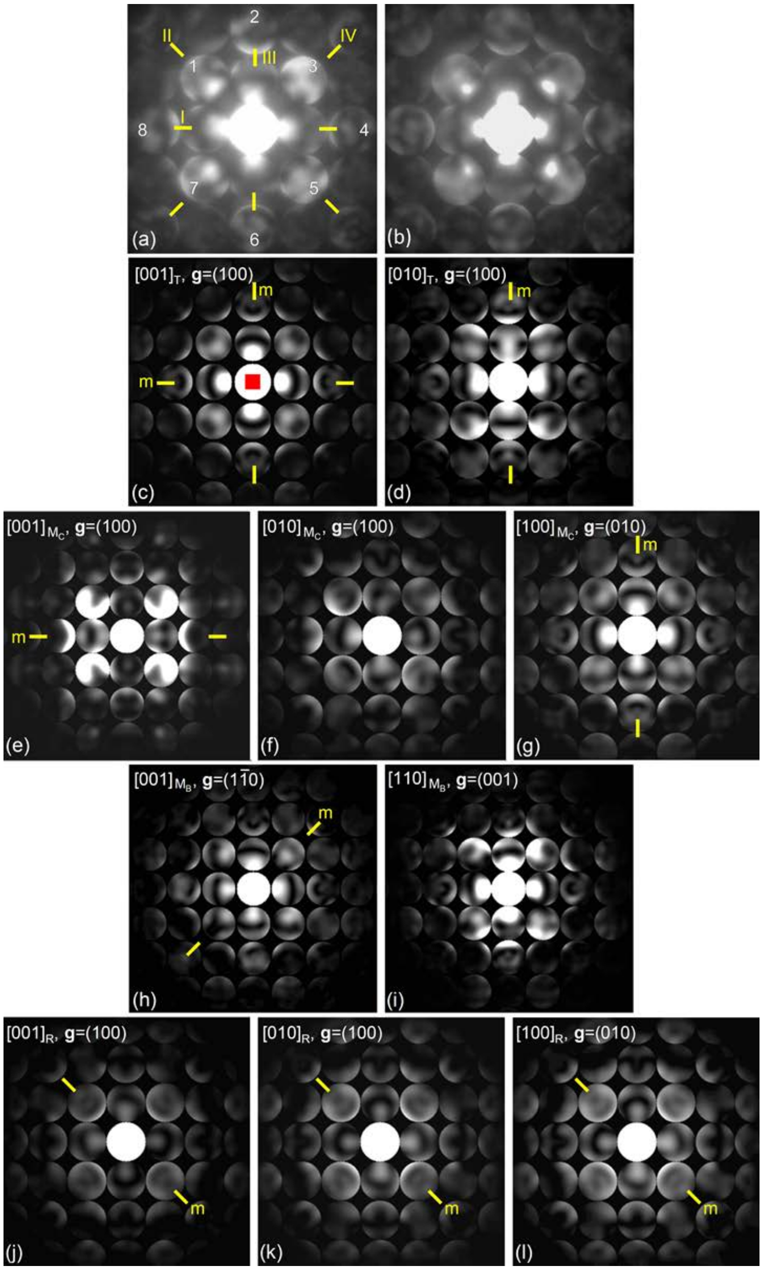


Figure 5.4: Experimental CBED patterns recorded along (a) $[001]_C$ and (b) $[010]_C$ and (c-l) simulated CBED patterns based on electron diffraction indexing results. The **g** refers to the first reflection along the horizontal direction.

The CBED patterns were then simulated for different thicknesses (t) based on the indexing determinations. The simulation results are best matched to the experimental patterns at t ~ 50 nm. The simulated CBED patterns based on the different structural models are shown in Figs. 5.4(c)-(l). Now, the observed symmetry in the experimental CBED pattern is compared with simulation results. In Fig. 5.4(c), the pattern symmetry of the T structure is 4mm along zone axis $[001]_T$. The simulation results for $[010]_T$, $[001]_{M_C}/[100]_{M_C}$, $[001]_{M_B}$ and $[001]_R/[010]_R/[100]_R$ CBED patterns have the mirror (m) symmetry element. No such mirror element was observed in the simulation patterns for $[010]_{M_C}$ and $[110]_{M_B}$ similar to the experimental CBED patterns. Based on the experimental and simulation results here, therefore, the closest matches with experiment are $[010]_{M_C}$ and $[110]_{M_B}$.

5.4.2.2. Local symmetry along $[011]_C$

Figure 5.5(a) shows a selected experimental CBED pattern for ZOLZ along zone axis $[011]_C$. The $[011]_C$ CBED patterns were measured for two possible mirror orientations along lines I and II in Fig. 5.5(a). The highest cross-correlation coefficient for mirror symmetry was obtained for line II with $\gamma_{m(II)}$=59%.

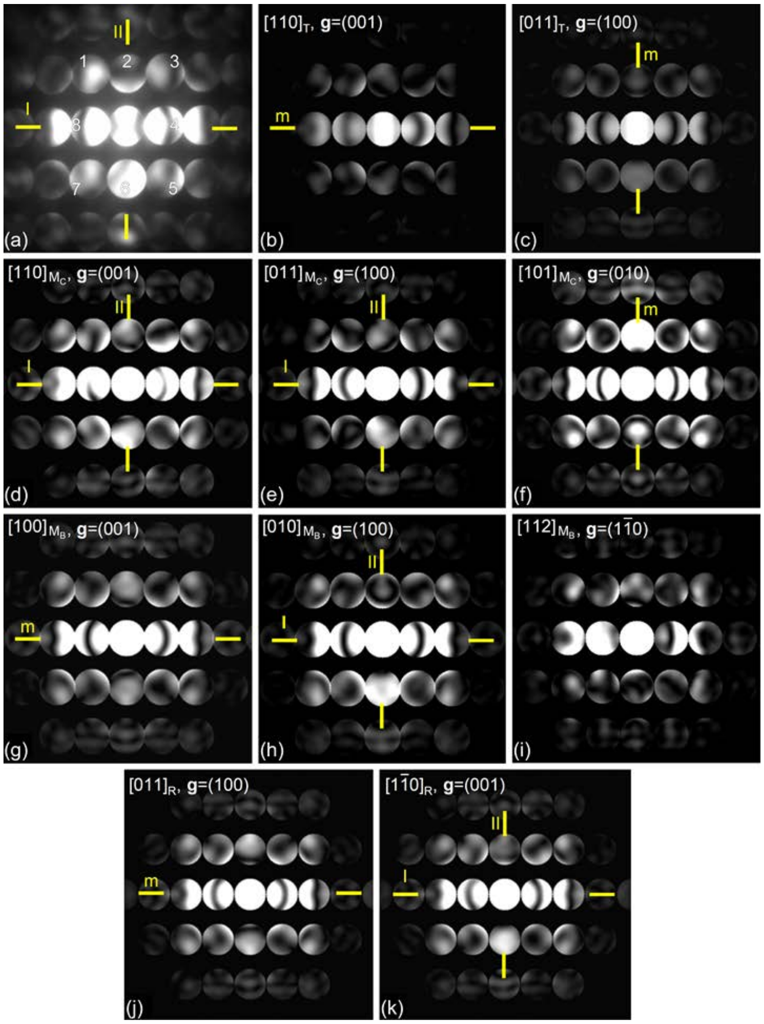


Figure 5.5: (a) Experimental CBED pattern recorded along $[011]_C$, and (b-k) simulated CBED patterns based on the electron diffraction indexing results. The **g** refers to the first reflection along the horizontal direction.

Figures 5.5(b)-(k) show the simulated CBED patterns corresponding to the indexing results. Based on the simulations, several indexing options can be directly excluded from possible solution for the $[011]_C$ zone axis. The simulated patterns for $[110]_T$ / $[011]_T$, $[101]_{M_C}$, $[100]_{M_B}$, and $[011]_R$ have mirror symmetry, which is inconsistent with the low cross-correlation coefficient obtained from experimental data. The simulated CBED pattern for $[112]_{M_B}$ is obviously different from the experimental pattern as shown in Fig. 5.5(i). Patterns for $[110]_{M_C}$ /

$[011]_{M_C}$, $[010]_{M_B}$ and $[1\bar{1}0]_R$ show no mirror symmetry like the experimental pattern. The amount of deviation from mirror symmetry in these simulated patterns was then measured along I and II in the same way as done previously for the experimental patterns. According to these measurements, $[110]_{M_C}$ is more mirror-symmetrical along line I ($\gamma_{m(I)}$=60%) compared with line II ($\gamma_{m(II)}$=38%), which is inconsistent with the experimental pattern. For $[011]_{M_C}$, significant breakdown for mirror symmetry occurred along lines I and II ($\gamma_{m(I, II)}$<30%). Simulated patterns for $[010]_{M_B}$ ($\gamma_{m(I)}$=16%, $\gamma_{m(II)}$=88%) and $[1\bar{1}0]_R$ ($\gamma_{m(I)}$=33%, $\gamma_{m(II)}$=72%) are more mirror-symmetrical along line II in agreement with experimental data. Based on these results, the possible matches here are for $[010]_{M_B}$ and $[1\bar{1}0]_R$.

5.4.2.3. Local symmetry along $[111]_C$

Figure 5.6(a) shows an experimental CBED pattern recorded along zone axis $[111]_C$, in which the first-order diffraction discs show 6-fold like features, but with significant symmetry breakdown in all discs. The breakdown is even more significant in the second and third-order diffraction discs. The amount of mirror symmetry was quantified by using the second- and third-order discs along the 8 possible mirror directions indicated in Fig. 5.6(a). The largest γ_m value is 26%, indicating a lack of symmetry, except for a trivial 1-fold rotation axis.

Figures 5.6(b)-(f) are the simulated CBED patterns (t=60 nm) corresponding to the indexing results for the $[111]_C$ zone axis. The $[111]_T$, $[101]_{M_B}$ and $[111]_R$ show higher symmetry than the experimental patterns and they are excluded from possible solutions for the zone axis of $[111]_C$. For further comparison, we quantified the mirror symmetry for $[111]_{M_C}$ and $[011]_{M_B}$ along the dotted line indicated, and the γ_m values were found to be 73% for $[111]_{M_C}$, and 95% for $[011]_{M_B}$, respectively. These γ_m values calculated from the simulations were significantly higher than those obtained from the experimental CBED pattern (< 26%).

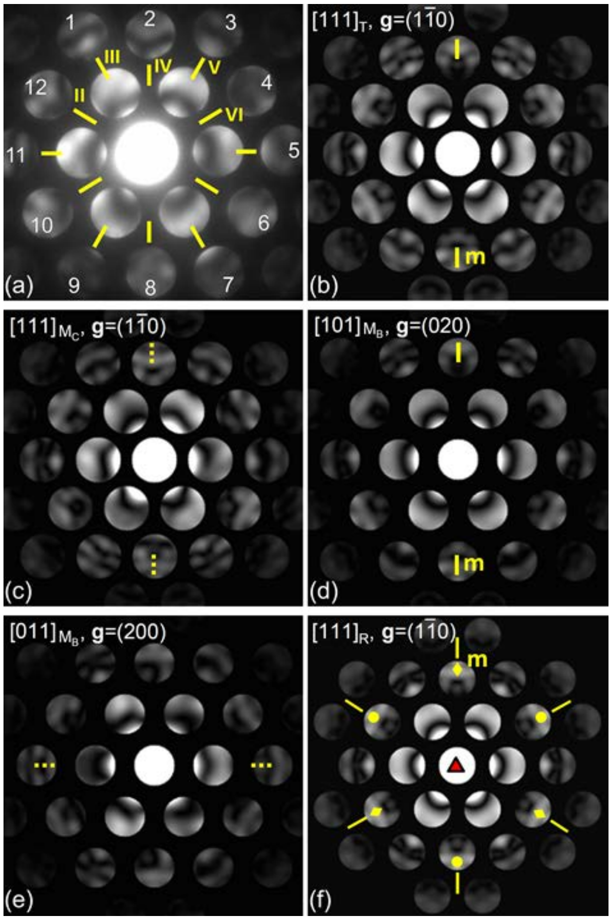


Figure 5.6: (a) Experimental CBED pattern, and (b-f) simulated CBED patterns for the selected crystal structures. The **g** refers to the first reflection along the horizontal direction.

5.4.3. Symmetry determination at tens of nanometers scale

The symmetry of CBED patterns was also investigated by using a larger electron probe of 35 nm (Fig. 5.7(a)). Based on the characteristic 10 nm size of nanodomains (Fig. 5.2), the probe shown in Fig. 5.7(a) would cover multiple nanodomains [56, 124, 125]. CBED patterns were obtained from different regions of the crystal along $[001]_C$, $[011]_C$ and $[111]_C$ zone axes. Figures 5.7(b)-(i) show selected experimental CBED patterns recorded along zone axes $[001]_C$, $[011]_C$ and $[111]_C$. The γ values for mirror symmetry were determined along the orientations indicated. The symmetry quantification results are summarized in Table 5.3. The lowest and highest γ_m

values measured from multiple CBED patterns are shown in the table. The experimental CBED patterns showing the maximum γ_m values were used for the symmetry determination.

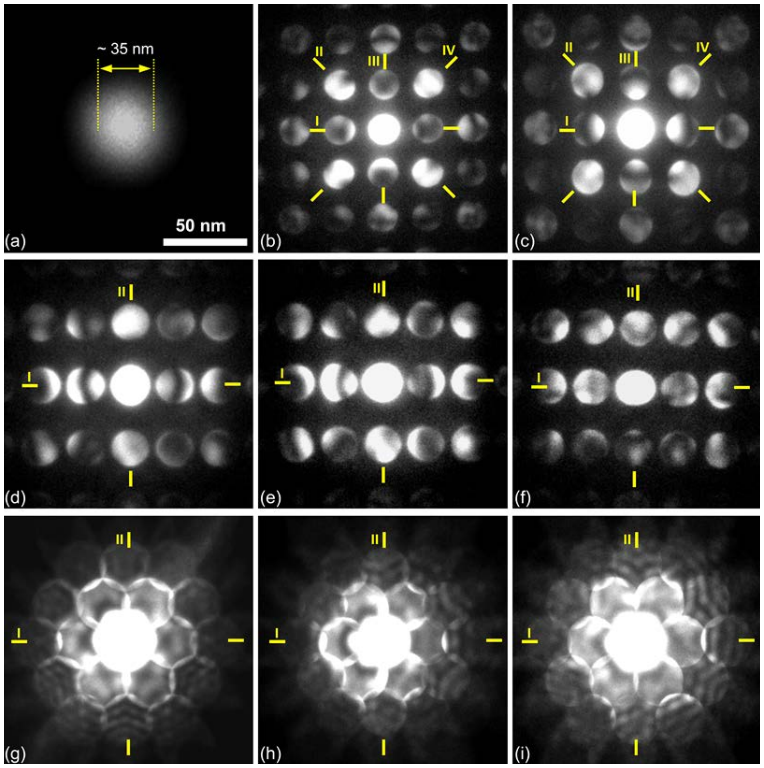


Figure 5.7: Selected CBED patterns recorded using the larger electron probe size of ~ 35 nm. The probe diameter is shown in (a). The experimental CBED patterns are shown for zone axes (b, c) $[001]_C$, (d, e, f) $[011]_C$ and (g, h, i) $[111]_C$. Mirror symmetry is quantified along the lines indicated in the CBED patterns.

For $[001]_C$, Figs. 5.7(b) and (c) show two selected patterns from different regions of the PMN-31%PT crystal. Figure 5.9(b) has the highest γ_m value of ~ 85% along line IV, while the other directions were less, 40% ~ 55%. For the second CBED pattern shown in Fig. 5.7(c), the calculated γ_m values ranged from 10% to 57%.

Table 5.3: Mirror quantification for $[001]_C$, $[011]_C$ and $[111]_C$ CBED patterns obtained with a probe size of 35 nm in diameter, and comparison with simulated CBED patterns for M_B symmetry. (The simulations are indicated by parentheses)

Zone axis	**CBED pattern**	**Mirror direction**	**γ_m (%)**	
			Exp.	M_B
$[001]_C$	Fig. 9(b) ($[001]M_B$, Fig. 10(a))	I	40	36
		II	55	19
		III	53	27
		IV	85	100
	Fig. 9(c) ($[110]M_B$, Fig. 10(b))	I	10	35
		II	21	37
		III	57	41
		IV	41	0
$[011]_C$	Fig. 9(d) ($[010]M_B$, Fig. 10(c))	I	36	29
		II	54	84
	Fig. 9(e) ($[112]M_B$, Fig. 10(d))	I	52	61
		II	9	20
	Fig. 9(f)	I	36	-
		II	32	-
$[111]_C$	Fig. 9(g) ($[101]M_B$, Fig. 10(e))	I	27	13
		II	83	100
	Fig. 9(h) ($[011]M_B$, Fig. 10(f))	I	77	96
		II	39	13
	Fig. 9(i)	I	39	-
		II	53	-

The CBED patterns recorded along $[011]_C$ (Figs. 5.7(d)-(f)) show different extents of possible mirror symmetry. Based on quantification results, the recorded patterns can be categorized into three categories. The first type (Fig. 5.7(d)) has a higher cross-correlation coefficient along line II ($\gamma_{m(II)}$=54%) than for line I ($\gamma_{m(I)}$=36%). The second type (Fig. 5.7(e)) is greater along line I ($\gamma_{m(I)}$=58%) than for line II ($\gamma_{m(II)}$=38%). The third type (Fig. 5.7(f)) has lower values of 36% and 32%, for lines I and II, respectively.

We also observed three types of CBED patterns for the $[111]_C$ direction as shown in Figs. 5.7(g), (h) and (i). The experimental CBED patterns have no 3m symmetry, so the R structure is directly ruled out. As listed in Table 5.3, the first type (Fig. 5.7(g)) has the highest cross-correlation coefficient for mirror symmetry along line II ($\gamma_{m(II)}$=83%), while the second type (Fig. 5.7(h)) has the highest value along line I ($\gamma_{m(I)}$=77%). The results for the third type (Fig. 5.7(i)) were significantly lower.

From the simulated patterns for $[001]_C$, $[011]_C$ and $[111]_C$ as described in Ch. 5.4.2, the Cm (M_B) symmetry provided the closest match with the zone axes investigated here. Figure 5.8 summarizes the results for CBED simulations based on Cm symmetry. For example, the simulated patterns in Figs. 5.8(a), (b), (c), (d), (e), and (f) are in close match with the experimental patterns shown in Figs. 5.7(b), (c), (d), (e), (g), and (h), respectively. The approximate mirror observed in Fig. 5.7(b) is consistent with the mirror element found in $[001]_{M_B}$ (Fig. 5.8(a)). The $[001]_C$ CBED pattern shown in Fig. 5.7(c) is comparable with $[110]_{M_B}$ shown in Fig. 5.8(b). For $[011]_C$, the $[010]_{M_B}$ and the $[112]_{M_B}$ simulations are in agreement with the experimental data given in Fig. 5.7(d) and (e), respectively. $[010]_{M_B}$ has a higher cross-correlation coefficient for more mirror-like symmetry along line II ($\gamma_{m(I)}$=29%, $\gamma_{m(II)}$=84%), and $[112]_{M_B}$ is more mirror-symmetrical along line I ($\gamma_{m(I)}$=61%, $\gamma_{m(II)}$=20%). For $[111]_C$, $[101]_{M_B}$ has perfect mirror symmetry while $[011]_{M_B}$ has approximate mirror about the dotted line with a γ_m value of 96%. The two selected $[111]_C$ CBED patterns in Figs. 5.7(g) and (h) are thus comparable with $[101]_{M_B}$ and $[011]_{M_B}$, respectively.

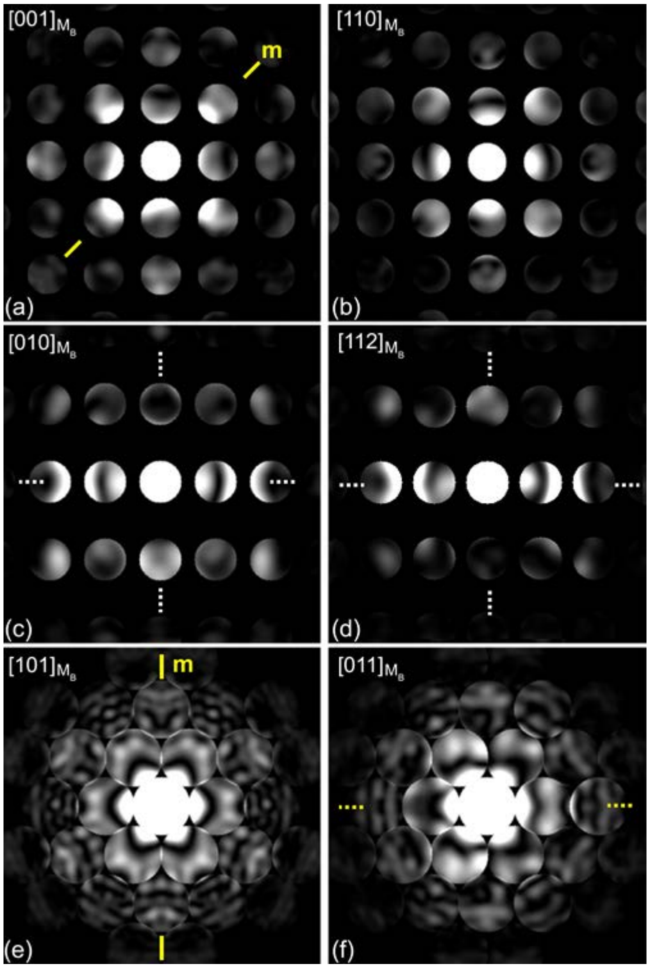


Figure 5.8: Simulated CBED patterns based on the structure of the M_B phase for zone axes (a) [001], (b) [110], (c) [010], (d) [112], (e) [101], and (f) [011]

5.5 Discussion

XRD determines the average symmetry at the length scale of X-ray coherence (hundreds of nanometers). From the XRD result, the strong peaks are for dominant, averaged, domain orientations, whereas the weak peaks are from domains of different orientations. Results for XRD indexing of single crystal data in Table 5.1 are close to monoclinic M_C (Pm) symmetry. Evidence of lattice differences with the reported data is observed in the strong diffraction peak positions. The experimental peak positions deviate slightly from the calculated peak positions by 0.07^o ~ 0.3^o. The difference in the peak positions is 0.15 ~ 0.2%. Overall, the XRD results suggest the PMN-31%PT crystal has monoclinic M_C structure with small lattice distortions from published data based on powder specimens.[51]

Approximate mirror symmetries were detected when probing with the larger 35 nm diameter electron beam. The experimental CBED patterns, with help of simulations, can be indexed with the Cm symmetry as shown in Ch. 5.4.3. However, the observed mirror symmetry in the experimental pattern is imperfect with the highest cross-correlation coefficients of γ_m = 83 ~ 85% compared with γ values of 96 to 100% for simulated patterns. Experimentally, we found that the mirror symmetry can be detected using a 15 nm, or larger, sized electron probe. The 35 nm probe used here is twice as larger than required for detecting the mirror symmetry element. Thus the averaged symmetry at 35 nm length scale is therefore concluded as a monoclinic Cm-like symmetry.

By comparison, when using the smaller 1.6 nm beam size, the cross-correlations for mirror symmetry are significantly less in the range of 0 ~ 42%, 10 ~ 66%, 8 ~ 59% and 0 ~ 26% for CBED patterns recorded along zone axes $[001]_C$, $[010]_C$, $[011]_C$ and $[111]_C$ respectively. These experimental CBED patterns resolve trivial symmetry of 1-fold rotation for all investigated zone axes. In the simulated CBED patterns, the monoclinic M_B and M_C predict different mirror symmetry along $[001]_C$, $[010]_C$, $[011]_C$, and $[111]_C$, dependent on the orientation of the monoclinic axes. For example, the experimental $[010]_C$ CBED pattern is expected to have either the symmetry of $[001]_{M_C}$ or $[100]_{M_C}$ based on the XRD indexing result. These possible mirror symmetries, however, were not detected by the 1.6 nm diameter probe size. R symmetry can be ruled out directly from the lack of 3m symmetry along $[111]_C$. Thus, from local symmetry investigations, significant deviations were observed from the symmetry of all reported structures.

Kisi et al. suggests that the observed monoclinic symmetry at the macroscopic scale is not a real phase but a symmetry induced by distortions from a residual stress or piezoelectric response [59]. In this case, the observed monoclinic structure can be considered as a result from the distorted R and T phases. Grinberg et al. predicted that the atomic displacements and local distortions varied with the local arrangement of B-site cations in PMN-PT system [126]. Suewattana and coworkers also examined lattice distortions in the PMN system [127]. Locally different atomic displacements have been experimentally determined by Egami, who argued that the local structure of relaxor ferroelectrics is different from the average structure obtained by conventional diffraction methods, such as Rietveld refinement of powder XRD data. Deviations from the reported symmetries at the few nm scale are clearly observed in all experimental CBED patterns. Locally-distorted structural models are therefore consistent with our observations by use of a 1.6 nm electron probe.

In contrast, the adaptive phase model proposes that the monoclinic symmetry is a result from the twin-related R or T nanodomains [54, 57, 61, 63]. We found no evidence of the T and R symmetry at the local scale (1.6 nm probe). Thus, the adaptive phase model is not consistent with our experimental observations.

On the other hand, the observed Cm-like symmetry is attained by averaging some regions around 35 nm scale. In other regions, CBED patterns without any symmetry, other than 1-fold rotation, are also observed using the same electron probe size. Hatch et al. proposed that averaging over the different volume fraction of domains may lead to different macroscopic symmetries [128]. Thus, the symmetry variations observed in the experimental CBED patterns may well reflect the change in local domains and their volume fractions.

The obtained solutions for the Cm-like structure were then confirmed by matching sample orientations by considering the relationship between the monoclinic unit cell axes and the pseudocubic axes for PMN-31%PT. According to the polarization rotation model, the monoclinic unit cell of M_B is rotated 45^o about the *c*-axis with respect to the pseudocubic *c*-axis [25, 26, 46]. The M_B axes of a_{M_B} and b_{M_B} are therefore along the pseudocubic axes of $[\bar{1}\bar{1}0]_C$ and $[1\bar{1}0]_C$. The monoclinic *c*-axis is tilted from the pseudocubic *c*-axis by β. Figure 9 schematically shows the monoclinic M_B unit cell with respect to the pseudocubic unit cell for different orientations. In the crystal, the $c_{\mathrm{M_B}}$ can be along any of the three pseudocubic axes of $[001]_C$ (Fig. 5.9(a)), $[100]_C$ (Fig. 9(b)) or $[010]_C$ (Fig. 5.9(c)) as confirmed by XRD. For the $[001]_C$, the M_B structure can be along <001> and <110>. For $[011]_C$, the corresponding M_B zone axes are $<010>_{\mathrm{M_B}}$, (or $<100>_{\mathrm{M_B}}$) and $<112>_{\mathrm{M_B}}$, and for $[111]_C$ they are $<101>_{\mathrm{M_B}}$ and $<011>_{\mathrm{M_B}}$. These relationships are consistent with the solutions of M_B shown in Fig. 5.8. Therefore, the solutions for the monoclinic Cm (M_B)-like symmetry are supported by not only the CBED simulations but also by the relationships with the crystal orientations.

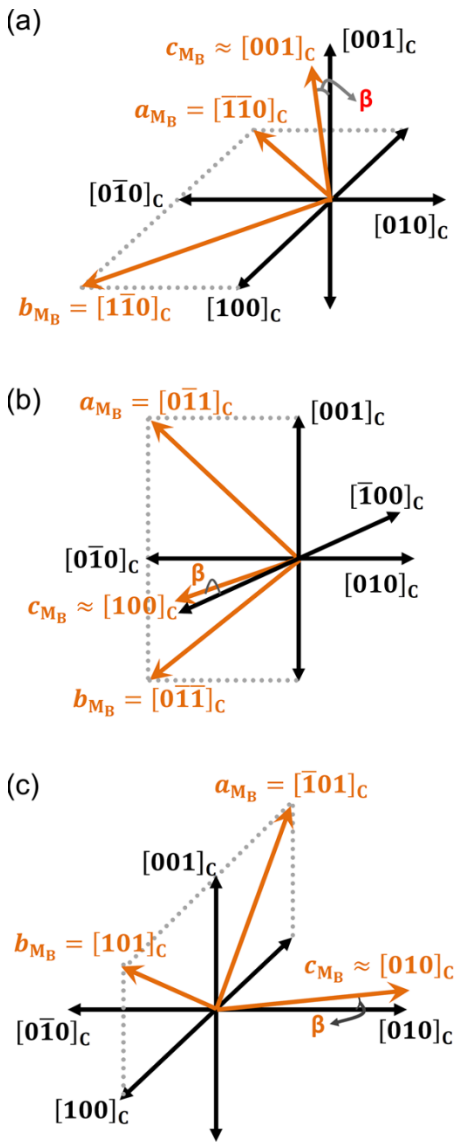


Figure 5.9: Orientation relationships between monoclinic M_B (Cm) and pseudocubic axes.

The presence of the diffraction peaks belonging to different lattice planes in a single crystal XRD pattern suggests that they come from different domains. The large difference in the width of the diffraction peaks between the strong and weak peaks suggests a difference in the average domain size. The size of domains measured by XRD, however, is on the order of hundreds of nanometers as measured by the strong diffraction peaks, which is much larger than the size of nanodomains observed in Fig. 5.2. On these length scales, it is likely that XRD averages over multiple domains consisting of the Cm-like symmetry. Whether this averaging leads to the symmetry of M_C, or XRD simply reflects the coexistence of phases, is not clear and requires further clarification.

5.6 Summary and conclusion

We have investigated the symmetry of PMN-31%PT single crystal from the local scale to the macroscopic scale by using XRD and CBED. The results show that the symmetry of PMN-31%PT is triclinic at a few nm in length scale, and becomes monoclinic Cm-like symmetry at the length scale of a few tens of nm. The macroscopic symmetry determined by XRD suggests multiple domains of different sizes in PMN-31%PT single crystal. Thus, the high piezoelectric response of PMN-31%PT single crystal at the MPB region is underlined by a structure that lacks local symmetry, which has an averaged monoclinic symmetry over tens of nanometers in some regions of the crystal. The lack of local symmetry may enhance the polarization switching behavior in this material. The monoclinic Cm-like symmetry provides, in addition, a structural bridge for collective polar rotation on the scale of tens of nm that is a key to the enhancement of the piezoelectric properties under external electric field. Our result is of importance to show the missing links between the local symmetry and macroscopic symmetry.

CHAPTER 6

NANOSCOPIC SYMMETRY AND POLARIZATION SPATIAL FLUCTUATIONS IN A RELAXOR-BASED FERROELECTRIC CRYSTAL[6]

Single crystals of Pb(Mg1/3Nb2/3)O3-31%PbTiO3 (PMN-31%PT) are known for their complex domain structures at the nanometer-scale. While their average symmetry has been studied by x-ray, neutron and electron diffraction methods, there is little knowledge about variations in symmetry at the local scale. Here, we provide direct evidence for the volume dependency and the spatial dependency of symmetry fluctuations by using quantitative convergent beam electron diffraction (CBED) and energy dispersive x-ray (EDX) spectroscopy. Fluctuations in symmetry were determined by using different electron beam probe sizes ranging from ~2 to 25 nm from a crystal ~62 nm thick. The symmetry of PMN-31%PT was found to increase linearly as the average volume increased, and the local symmetry fluctuated from one location to another at the nanoscale. The EDX spectroscopy results indicate the symmetry fluctuations are strongly correlated with significant changes in chemical composition at the local scale in volume of few tens nm3. We attribute the symmetry fluctuation with locally varying polarization.

6.1 Introduction

It is usual to determine symmetry from a volume element of material. For the case of x-ray or neutron diffraction studies, the symmetry of crystals is averaged and filtered by Bragg diffraction using probes ranging from microns to mm. A new structure was determined by such methods for the complex ferroelectric (1-x)$Pb(Mg_{1/3}Nb_{2/3})O_3$-x$PbTiO_3$ (PMN-xPT) at the morphotropic phase boundary (MPB) [25, 49, 67]. This new monoclinic (M) phase was proposed

---

[6] This chapter is submitted to Journal of Applied Crystallography.

to be a structural bridge that would facilitate polarization rotation from the rhombohedral (R) to tetragonal (T) states across the MPB [45, 46]. On the other hand, local chemical fluctuations are expected based on the substitution of B-site cations. Chemical fluctuations induce local distortions that are governed by the nature of chemical bonds. Indeed, neutron total diffraction measurements revealed atomic pair distances that deviated significantly from that of the averaged crystal structure [129, 130]. However, the extent of chemical fluctuations in real crystals is poorly understood and the development from short-range disordered structure to long-range ordered structure with well-defined symmetry is unknown. This knowledge gap has led to considerable debates about the nature of the M phase in the MPB region[58, 63, 100] and the origin of the giant piezoelectric coefficients that is observed in these relaxor (PMN)-based ferroelectric (PT) materials in general [14, 59].

A PMN-PT single crystal is known for having complex nanodomain structures. TEM images reveal tweed-like contrast, which is common to relaxor-based ferroelectrics and complex oxides in general [100, 124, 131]. The width of tweed-like contrast is on the order of a few tens of nm. Thus, symmetry analysis of nanodomains requires a nanometer sized probe, which can be achieved with the convergent beam electron diffraction (CBED) technique [71, 85, 117, 132]. Applications of CBED analysis to $Pb(Zr_{1-x}Ti_x)O_3$ (PZT) and PMN-PT systems have revealed the evidences of the mirror symmetry associated with the R, M and T phases in these crystals [56, 64, 65, 100, 104, 119, 133, 134]. Nevertheless, there are several drawbacks associated with these earlier CBED symmetry analyses. First, the sample history was not carefully considered even though the state of polarization depends sensitively on external forces applied to the sample during the sample preparation [96]. Second, interpretation of CBED pattern symmetry is rather arbitrary based on visual inspection so the extent of the observed symmetry could not be quantified [107, 112]. Last but not the least, all previous CBED symmetry analyses of complex ferroelectric crystals only provide the averaged symmetry at the selected sample areas. Because of the complex domain structure, the symmetry is expected to vary across the sample on the scale of the observed nanodomains. However, such information is not yet available.

Recently, Kim *et al* reported that the detection of the M (*Cm*)-like symmetry in the *unpoled* PMN-31%PT single crystal depends on the averaged volumes [135]. The M (*Cm*)-like symmetry was only detected with a probe of 35 nm in diameter with 85% mirror to the perfect mirror. Thus, the local symmetry of PMN-31%PT single crystal is not only position dependent, but also volume dependent.

Motivated by the above considerations, we report here direct measurements of local symmetry fluctuation as a function of different volumes and local areas. The measurement was made possible by using a new technique of scanning CBED combined with symmetry quantification [107]. The local symmetry is averaged over a volume defined by the electron probe size (~2 to 25 nm) and the crystal thickness (~60 nm). By quantifying the symmetry recorded in CBED patterns, we show that the symmetry in PMN-31%PT is linearly related to the averaged volume. The local crystal symmetry fluctuates, and the observed fluctuation is shown to be probe-size dependent, decreasing significantly when the probe size increases above ~10 nm. The origin of these fluctuations is studied further by examining the symmetry of different diffraction orders and chemical fluctuations within different volumes and locations using energy-dispersive X-ray (EDX).

6.2 Experimental methods

For the experimental measurement, a PMN-31%PT single crystal was selected from a melt-grown crystal. The crystal was sliced normal to $(111)_C$ (subscript 'C' refers to the cubic-axes) and then thinned for electron transparency using procedures described previously [96]. In $(111)_C$ orientation, point symmetries of *3m*, *m* and *m* are expected for R, T and M_B (*Cm*) space groups. The prepared TEM specimens were further annealed at 500 °C in air with slow cooling to remove ion-milling induced artifacts and induced domain structures [96]. Figure 1 shows the structural evolution for the different sample history studied by x-ray diffraction, which indicates the evolution of domain structure. The x-ray profiles were recorded from a PMN-31%PT TEM specimen subjected to thermal annealing (Fig. 6.1(a)), ion-milling (Fig. 6.1(b)) and poling (Fig. 6.1(c)) [001]$_C$. The annealed sample was respectively ion-milled and poled along [001]$_C$ at ~15 kV/cm for the XRD profiles as shown in Fig. 6.1(b) and (c). The annealing process depoles the sample so that the annealed sample can be considered as unpoled. As shown in Figs. 6.1(b) and (c), the ion-milled and the poled sample show the significant changes in the magnitude of 2θ peaks at ~21.9, ~44.8, 69.7, and ~99.2 from the annealed (unpoled) sample. The XRD profiles of ion-milled and poled sample are similar. Based on the reported crystallographic information of PMN-31%PT [51, 52], the ion-milled and the poled samples are close to the tetragonal structure (*P4mm*) while the annealed sample is close to the monoclinic (*Pm*) structure. We have selected the annealed sample for further study in order to examine the polarization in the unpoled state.

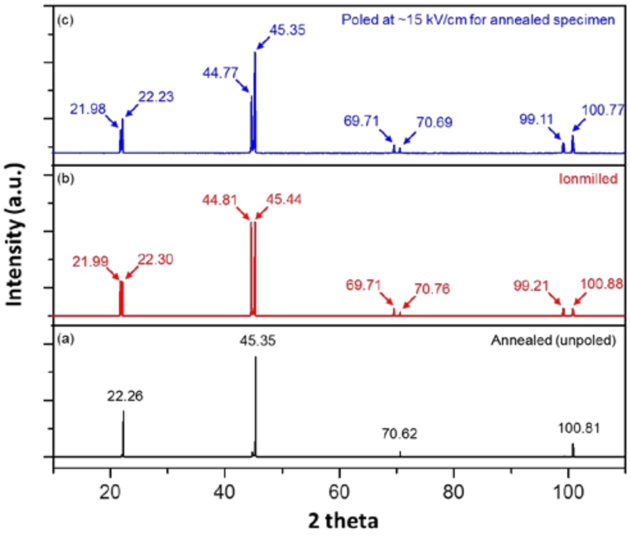


Figure 6.1: Recorded x-ray profiles for (a) an annealed, (b) ion-milled and (c) poled single crystal TEM specimen. For the poled sample, the annealed TEM specimen was subjected to a poling field at ~15 kV/cm applied along [001]$_C$.

CBED patterns were recorded using different probe sizes to measure the local crystal symmetry. To form focused electron probes of different sizes, we used a JEOL 2100 LaB_6 (JEOL, Tokyo, Japan) in the CBED mode and changed the demagnification of the condenser 1 lens (spot size). Using this approach, we were able to obtain electron probes from 2.7 to 25 nm in full-width at half-maximum (FWHM). The amount of mirror symmetry in the CBED patterns was then quantified using the normalized-cross correlation coefficient (γ) between two symmetry related CBED disks described in Ch. 4.

6.3 Results and discussion

6.3.1 Probe size-dependent symmetry variations

Figures 6.2(a-e) show the CBED patterns recorded from $[111]_C$ PMN-31%PT using different probe sizes of 2.7, 5.2, 9, 15, and 24 nm, respectively. The CBED patterns were recorded from the same region within the crystal. The patterns recorded for the 15 nm and 24 nm probes appear to have the mirror symmetry element (*m*) along the dotted-line indicated in Figs. 6.2(d) and (e). This mirror symmetry, however, is *not* observed in the patterns recorded using smaller probe sizes less than ~9 nm. For comparison, we also recorded CBED patterns from a Si single crystal along the $[110]_C$ zone orientation using similar electron probe sizes. These patterns are also shown in Figs. 6.2(f-j). By contrast with PMN-31%PT, *2mm* whole pattern symmetry was preserved in Si for the different probe sizes ranging from 2.7 to 25 nm, whereas, the patterns obtained from the ferroelectric relaxor were size dependent.

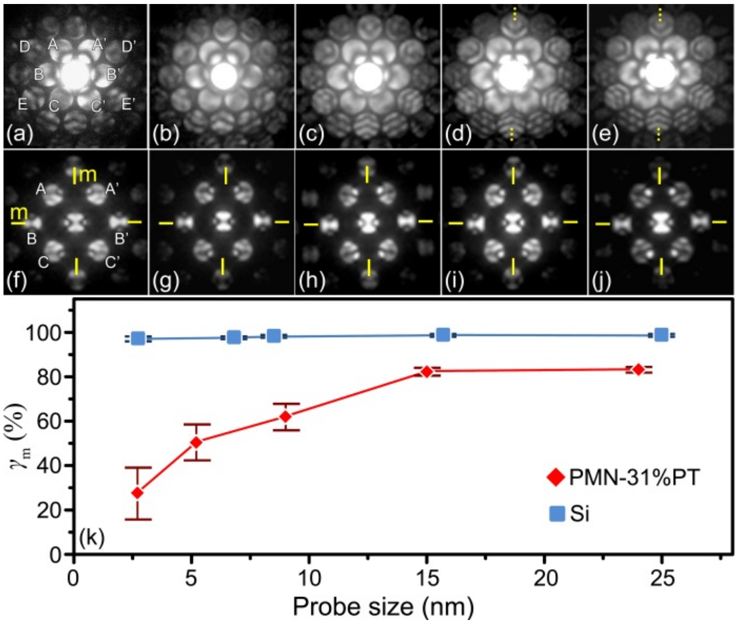


Figure 6.2: Variations in CBED patterns obtained with different probe sizes ranging from 2.7 nm to 25 nm for (a-e) PMN-31%PT and (f-j) Si single crystal. The cross-correlation coefficient for the mirror element (γ_m) is quantified along the indicated direction and plotted in (k). The error bars reflect symmetry variations obtained from measurements on different CBED patterns.

Figure 6.2(k) illustrates the γ_m variations as a function of probe size. The γ_m values were quantified using the (A/A')/(B/B')/(C/C')/(D/D')/(E/E') and (A/A')/(B/B')/(C/C') disc pairs for the PMN-31%PT single crystal and the Si single crystal, respectively. The error bars plotted are the standard deviations measured from multiple CBED patterns recorded from different areas of similar thickness (~65 nm). The PMN-31%PT $[111]_C$ CBED pattern has the lowest γ_m value of 27.3% at 2.7 nm, and rises thereafter to about 83.1% as the probe size increases above 9 nm. By contrast with PMN-31%PT, the γ_m values for Si $[110]_C$ were almost constant at ~98% for all probe sizes used. Thus, unlike silicon, where the symmetry of the pattern was almost perfect and constant down to ~2.7 nm, the mirror symmetry of PMN-31%PT was only obtained by averaging over regions greater than ~15 nm. The highest symmetry obtained for PMN-31%PT was ~83% in cross-correlation coefficient, so the mirror symmetry recorded in the CBED patterns was only approximate for the larger probe sizes.

6.3.2 Spatial symmetry fluctuation and the principle of volume averaging

To further examine the local symmetry fluctuation as indicated by the error bars in Fig. 6.2(k), we recorded CBED pattern across a sample area using our scanning CBED technique [107]. The information in the recorded patterns provides a map of symmetry variations. Figure 6.3 shows the results obtained from a 45×15 nm^2 sample area inside the boxed region indicated in Fig. 6.3(a). The symmetry mapping was obtained over the same area using two different probe sizes of ~5 nm and ~15 nm. For the ~5 nm beam probe, CBED patterns were recorded from 9×3 grids with a step length of 5 nm as shown in Fig. 6.3(b). For the ~15 nm sized probe, CBED patterns were recorded from 3×1 grids with a step length of 15 nm as shown in Fig. 6.3(c). The mirror symmetry was then quantified along the same direction used in Fig. 6.2(a). Figure 6.3(b) shows the symmetry map for the 5 nm probe. The γ_m values ranged from 29.1% ~ 62.4% ($\gamma_{m(average)}$=48.5%), resulting in significant contrast variation in the symmetry map. In contrast, the symmetry map obtained with ~15 nm probe (Fig. 6.3(c)) showed an almost constancy in γ_m with an average value of 83.5% (σ(γ_m)=1.8%).

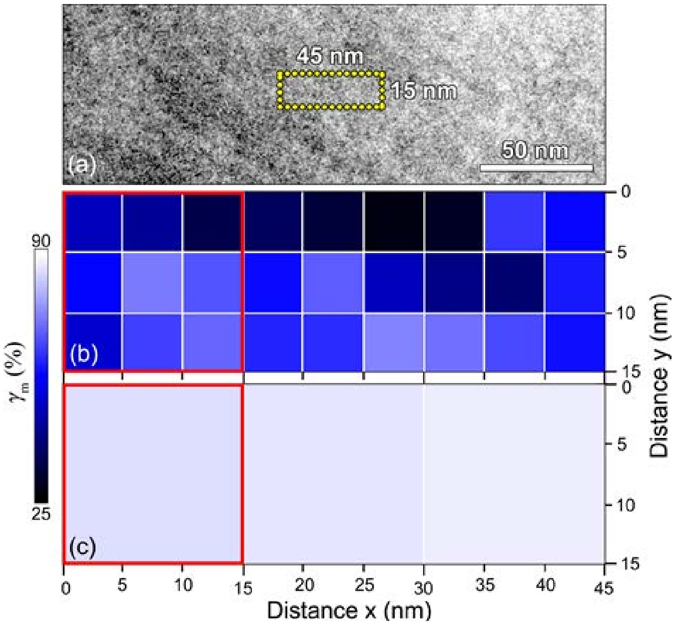


Figure 6.3: Symmetry maps obtained from the boxed area in (a) (45x15 nm^2) recorded for different probe sizes of (b) ~5 nm and (c) ~15 nm

This effect of volume averaging on symmetry is further examined. The experimental CBED patterns were extracted from the 9 grids inside the box marked in Fig. 6.3(b) as shown in Fig. 6.4(a). Values for γ_m varied from 41.9% to 60.8% for these patterns. Figure 4(b) is a composition of the 9 CBED patterns. For comparison, Fig. 6.4(c) shows the experimental CBED pattern extracted from the 1st grid point in Fig. 3(c), which is the same area given in Fig. 6.3(b). By contrast with the individual CBED patterns obtained with the 5 nm probe, the averaged pattern increased in γ_m value up to 84%, close to the experimental γ_m value of 82% recorded with ~15 nm probe.

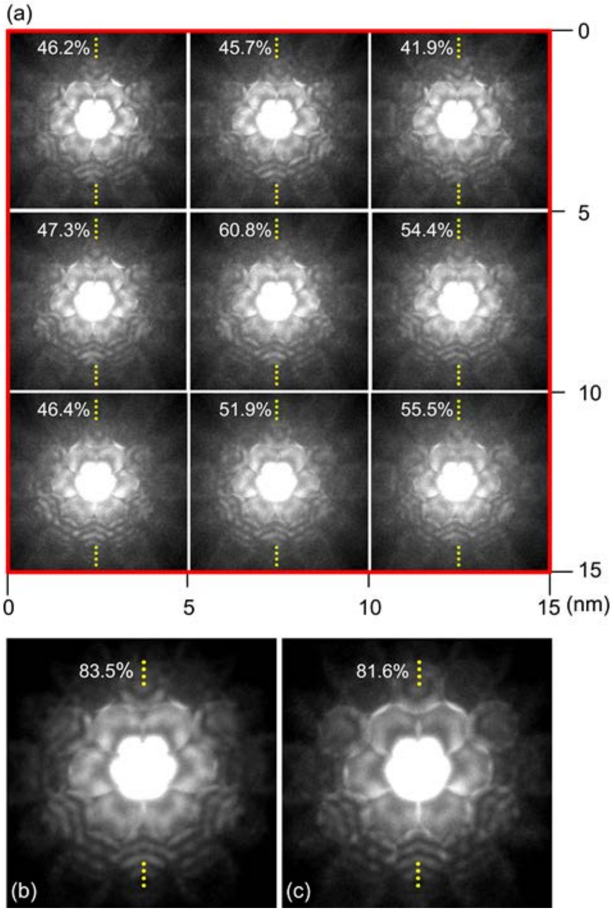


Figure 6.4: (a) Experimental CBED patterns extracted from the boxed area in Fig. 6.3(b). The extracted CBED patterns were merged into (b) an averaged CBED pattern in order to compare to (c) an individual CBED pattern for the larger ~15 nm probe. The γ_m values are indicated along the dotted line. The dotted line in CBED patterns is originally orientated at ~49° to the horizontal direction.

6.3.3 Nature of symmetry breaking: Chemical inhomogeneity and symmetry fluctuations

The origin of symmetry fluctuations can be considered in terms of chemical heterogeneity. The effect of chemical heterogeneities on local structures in relaxor ferroelectrics was previously suggested by Cross ,who proposed polar nano-regions (PNR) [136]. Unlike silicon (Si), which is chemically homogeneous, PMN-xPT is chemically heterogeneous, with multiple cations ($Mg^{2+}/Nb^{5+}/Ti^{4+}$) occupying B-sites in the perovskite (ABO_3) structure [137]. Thus, the possibility of chemical heterogeneities may occur for charge neutrality conditions with multivalent B-site cations.

To examine further whether the symmetry fluctuations observed in our experiments could be correlated with chemical fluctuations, we used a combination of statistical simulations and *in-situ* EDX spectroscopy techniques. The experimental result was compared to statistical calculation. For calculation, we modeled a volume element of 50×50×160 unit cells along $[100]_C$, $[010]_C$ and $[001]_C$ directions, respectively. The modeled unit cell contains a single perovskite unit with $a \approx b \approx c \approx 4$ Å. Thus, the total volume of the model was $20\times20\times64$ nm^3. The thickness of 64 nm was decided upon because the experimental CBED patterns match with simulations in the thickness range from 62 to 66 nm. The B-site cations of $Mg^{2+}/Nb^{5+}/Ti^{4+}$ are then randomly distributed into the modeled volume in the ratio of $Mg^{2+}:Nb^{5+}:Ti^{4+}=23:46:31$ for x = 0.31. From the modeled volume, the number of B-site cations was counted based on the volume defined by the electron probe size from 0.4 nm ~ 20 nm. The calculation was repeated multiple times to calculate any variations in B-site cation occupancy. For the EDS analysis, we calculated the atomic percent of $Mg^{2+}/Nb^{5+}/Ti^{4+}$ by changing the probe size from 1.4 to 22.3 nm. The EDS profiles were recorded at multiple positions (at least 10 different sites) for each probe size. Chemical results are illustrated in Fig. 6.5 together with the symmetry variation from Fig. 6.2(k). Calculations identified increasing fluctuations in chemical composition as the probe size decreased below 5 nm. For a 0.6 nm probe (i.e., approximately 1.5 unit cells in diameter) the $Mg^{2+}/Nb^{5+}/Ti^{4+}$

composition had standard deviations of 3.3%/3.6%/3.5%, respectively, as obtained from calculation results. Chemical fluctuations were found to approach zero (~0.1%) when the probe size was above 9 nm. Experimental EDX results showed a similar trend with larger probe sizes. Fluctuations in $Mg^{2+}/Nb^{5+}/Ti^{4+}$ were up to 3.4%/4.3%/3.0% at 1.4 nm probe size, and decreased to less than 1% (and remained approximately constant thereafter) for measurements with a probe size of 8.5 nm or larger. The 1% value can be attributed to the uncertainty in the EDX analysis, which limits the smallest amount of chemical fluctuations that can be detected experimentally. Overall, significant chemical heterogeneity could be observed and calculated when the probe size was ~5 nm or less, i.e., chemical fluctuations in nano-regions. In similar to the chemical fluctuation, the symmetry variation has the maximum at the probe size of 2.6 nm and drastically decreases as the probe size is bigger than 8 nm.

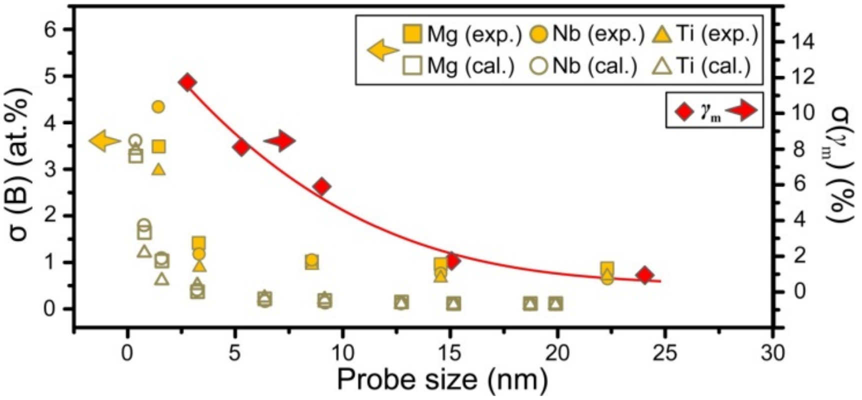


Figure 6.5: Open rectangle, circle and triangle are the statistically calculated chemical fluctuations for Mg^{2+}, Nb^{5+} and Ti^{4+}. The experimental EDS results are represented by solid rectangle, circle and triangle symbols. Variations in mirror symmetry $\sigma(\gamma_m)$ are indicated by the solid rhombus.

The relation between the chemical heterogeneities and the symmetry fluctuation is further examined. Chemical heterogeneities and local ionic displacements have different effects on crystal symmetry. Compositional fluctuations introduce changes in atomic scattering factors, which do not break the symmetry of CBED patterns. In PMN-PT, the A-site Pb^{2+} cation has lonepair $6s^2$ electrons and a tendency to off-center within the close-packed oxygen structure. According to theoretical predictions [126], the actual displacement of Pb^{2+} depends on the nature of the coordinated B-site cations of $Mg^{2+}/Nb^{5+}/Ti^{4+}$. Smaller ionic displacements for the B-site cations are also expected based on first principles theory. Experimental evidence for the correlated local ionic displacements has been reported in PMN and PMN-PT system using neutron total diffraction and synchrotron X-ray measurements [138, 139]. Thus, the chemical fluctuation induces the different averaged symmetry as a function of volume and affects the symmetry variation shown in Figs. 6.2-5.

The effect of ionic displacements can be directly evidenced by electron diffraction. According to Krivoglaz [140], the Bragg diffraction of a solid solution is treated from an ideal crystal by introducing an average structure factor $F_g = \sum_i \langle f_i \rangle \exp(2\pi i \mathbf{g} \cdot \mathbf{r}_i)$. In PMN-xPT, the reflections contained in the $[111]_C$ zone axis are dominated by the contribution from the Pb ion, which has the average atomic scattering factor of

$$\langle f_{\mathrm{Pb}} \rangle = f_{\mathrm{Pb}} \exp\left(-M_{\mathrm{Pb}}\right) = f_{\mathrm{Pb}} \left\langle \exp\left(2\pi i \mathbf{g} \cdot \Delta\mathbf{r}_{\mathrm{Pb}}\right) \right\rangle \quad (6.2)$$

The averaging of Eq. 6.2 gives the well-known, non-symmetry breaking result of $\exp\left(-M_{\mathrm{Pb}}\right) = \exp\left(-2\pi g^2 \left\langle \Delta r_{\mathrm{Pb}}{}^2 \right\rangle\right)$, which can be measured quantitatively by diffraction [141]. For the case when local ionic displacements do not average to zero, e.g., for spontaneous polarization $\mathbf{P_S}$, $\Delta\mathbf{r}_{\mathrm{Pb}}$ can be expressed as $\Delta\mathbf{r_S} + \Delta\mathbf{r_R}$ in the presence of local chemical fluctuation; one is systematic (S) and one is random (R). Their separation leads to

$$\langle f_{Pb} \rangle = f_{Pb} \exp\left(2\pi i \vec{g} \cdot \Delta\vec{r}_S\right) \exp\left(-2\pi g^2 \left\langle \Delta r_R{}^2 \right\rangle\right) \quad (6.3)$$

, where Δr_R should have a Gaussian distribution and average to zero. Thus, the effect of $\Delta \mathbf{r}_R$ on symmetry breaking can be ignored. To break the mirror (*m*) symmetry element, $\Delta \mathbf{r}_S$ must be out of the mirror plane, since $\mathbf{P}_S$ is somewhere within the mirror plane for the ideal crystal. Equation 3 predicts the amount of symmetry breaking increases with **g**. To examine the scattering angle dependence, we calculated the γ_m values given in Fig. 6.2(k) for the 1st ($\{110\}_C$ reflections) and 2nd order disks ($\{112\}_C$ reflections) separately. The results are shown in Fig. 6.6, where the 1st-order γ_m drops from 83.7% to 35.9%, and the 2nd-order γ_m drops from 83.1% to 14.4%, with decreasing probe size from 24 to 2.7 nm. Figure 6.6 is thus the evidence for the existence of ionic displacements in the PMN-31%PT single crystal. In addition, it can be shown that the ionic displacement is also averaged out as the averaged volume increases.

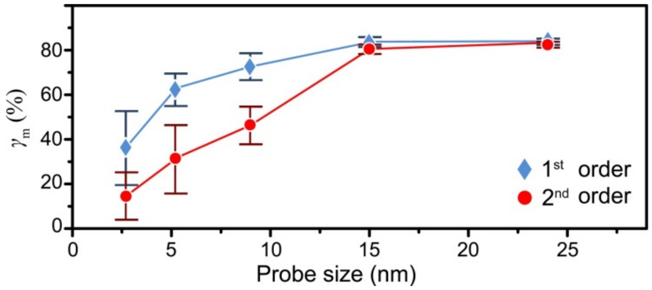


Figure 6.6: Development of mirror symmetry with increasing probe size obtained from the 1st and 2nd order disks

6.3.4 Local polarization fluctuations

The mirror-like symmetry in CBED patterns when recorded using probes 15 nm or larger approximately belongs to space group *Cm* [112]. Figures 7(a) and (b) compare the recorded CBED pattern (~15 nm probe) with simulations based on the Bloch-wave method and spherical ions. According to structural data [52], the spontaneous polarization direction is along the [-u0w] ([-0.25, 0, 0.03]), which is within the mirror plane as indicated by the arrow in Fig. 7(b). The above results show that monoclinic domains with an average polarization are detected in PMN-31%PT when using probes 15 nm or larger, while significant break down in symmetry is observed when using probes below 15 nm.

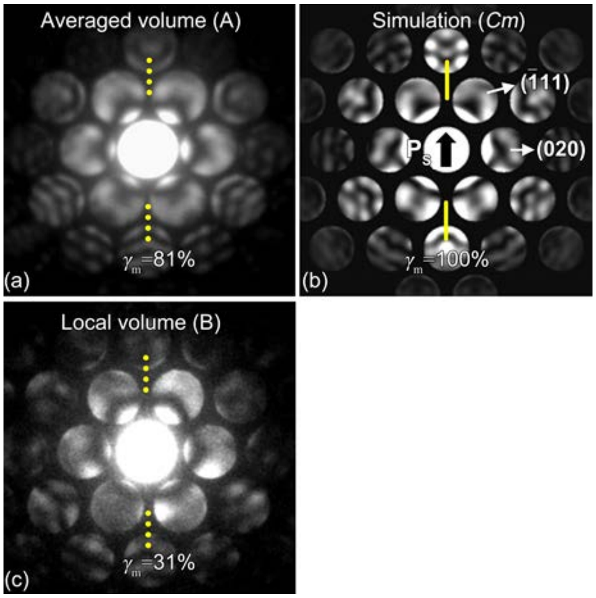


Figure 6.7: CBED pattern (a) obtained with the larger probe (~15 nm) is close to (b) monoclinic *Cm*, while (c) the symmetry detected with the smallest probe (~2 nm) is significantly broken.

The breakdown of symmetry can be attributed to local ionic displacements as shown in Eq. (3) and Fig. 6. Thus, the symmetry map (Fig. 3) can be taken as direct evidence for spatial fluctuations in local polarization, i.e., spontaneous polarization. Fluctuations in $\mathbf{P}_S$ increases as the probe size decreases. Large amount of symmetry fluctuation are observed when the electron probe was less than 10 nm. In comparison, significant chemical inhomogeneities were detected by EDX spectroscopy (Fig. 5) only when the probe size approaches 2 nm. Thus, the symmetry fluctuation cannot be attributed to chemical fluctuation alone. The extent of spatial symmetry fluctuation is on the scale of nanodomain size observed in this material. On the other hand, when using larger-sized probes, the fluctuations in symmetry (or $\mathbf{P}_S$) are averaged out, and the averaged symmetry is *Cm*-like.

6.4 Summary and conclusion

In conclusion, we have demonstrated the following new points: 1) local symmetry fluctuations indicate the existence of spatially varying polarization, 2) the crystal has macroscopic high symmetry (*Cm*-like) when averaged over a larger volume, and 3) chemical fluctuations are strongly correlated with the symmetry fluctuations and local polarization. This study, therefore, provides direct experimental evidence, and a possible critical link, between two previously disparate models for relaxor-based ferroelectrics, namely, the model of local polar nano-regions, and the polarization rotation model bridged by a separate monoclinic phase.

CHAPTER 7

IMAGING OF NANODOMAIN SYMMETRY AND POLARIZATION DIRECTIONS IN RELAXOR-FERROELECTRIC CRYSTAL AT THE MORPHOTROPIC PHASE BOUNDARY

Relaxor-ferroelectric single crystals, such as (1-x)Pb($Mg_{1/3}Nb_{2/3}$)O_3-x$PbTiO_3$ (PMN-xPT), have extraordinary high piezoelectric coefficients (d_{33}>2,500 pm/V and k_{33}>0.9) at the morphotropic phase boundary (MPB) [14] and the potential to transform the technology for medical imaging, actuation, and sensor. However, understanding the structure and properties in these crystals has been extremely difficult because of their complex chemistry and hiearachical structure across a range of length scales [55]. It was suggested that polarization rotation in a monoclinic phase at the MPB region driven by external electric fields was responsible for the large piezoelectric response [45]. While evidences for monoclinic symmetry have been reported by x-ray and neutron diffraction studies, the nature of this symmetry is a subject of debate for the lack of local probes [25, 26]. Here, we report a quantitative measurement and mapping of local symmetry and the related polarization using a newly developed scanning convergent beam electron diffraction (CBED) technique at spatial resolutions from ~1 to tens of nanometers. The monoclinic, M_B, symmetry is detected when the electron probe is larger than 10 nm in an annealed PMN-xPT single crystal at the MPB composition (x=0.31). Symmetry mapping shows 60° polarization domains of few tens of nanometer in width and regions without obvious monoclinic symmetry. The symmetry variation correlates with the tweed-like image contrast, whose origin for long has been a mystery. The symmetry is only close to monoclinic *Cm*, while the symmetry is not perfect. These observations suggest complex polarization patterns in PMN-31%PT where local averaging in some regions can lead to the observed monoclinic symmetry.

7.1 Introduction

The nature of spontaneous polarization in the solid solutions of relaxor and ferroelectric perovskites, such as PMN and PT, near the so-called MPB has been widely debated. In PMN-xPT at room temperature and low x, the polarization is along $\{111\}$ of rhombohedral (R) crystal symmetry (R3m), and changes to $\{001\}$ with tetragonal (T) symmetry (P4mm) at higher x. The MPB separates the R and T symmetries and defines a chemically (x)-driven change in morphology accompanied by polarization rotation[8]. Large enhancements in ferroelectric and piezoelectric properties [14] are observed in compositions near MPB. In PMN-xPT with x near the MPB composition (30 ~ 32%), x-ray diffraction results revealed multiple co-existing structures at low temperatures that can be indexed by monoclinic (M) symmetry [49, 66, 67]. The presence of monoclinic phase(s) provides a structural bridge that facilitates polarization rotation from the R to T phase [25]. However, the nature of the monoclinic symmetry has been hotly debated [58, 59, 126]. Grinberg et al. using ab. initio density functional theory demonstrated atomic displacements and local distortions that varied with the local arrangement of B-site cations [126]. Kisi et al. suggested that the observed monoclinic symmetry at the macroscopic scale is not a real phase but a symmetry induced by piezoelectric distortion or local residual stress [59]. It has been also suggested that the observed symmetry is a result of averaging over twin-related rhombohedral or tetragonal nanodomains [58].

What is clear from previous experiment is that these materials are spatially inhomogeneous. Coherent X-ray diffraction[142] provided evidences of polar nano-regions (PNRs) of ~10 nm in dimension about T_m in PMN-28%PT, while imaging using TEM and piezoelectric force microscopy revealed complex domain structures with scale ranging from tens of nm to μm[136],[143]. It is therefore critical to determine the local symmetry and its correlation with polarization in order to resolve the true nature of symmetry in PMN-xPT and complex ferroelectric crystals in general. For this purpose, we have selected a PMN-31%PT single crystal for study, whose composition falls inside the MPB region.

7.2 Results

We applied the scanning CBED technique to the study of single crystal PMN-31%PT (see details in Ch. 4). The crystal was annealed after sample preparation in order to observe its native symmetry (details of sample preparation are described in the method section). Figure 7.1(a) shows a bright filed (BF) image of an annealed PMN-31%PT single crystal along the $[111]_C$ with tweed-like contrast that has been reported before[96, 124, 131] (the subscript 'c' refers to the pseudo-cubic axis). The scanning CBED measurement was performed for the dotted box given in Fig. 7.1(a). We used an electron probe of 14.3 nm in FWHM (Full width half maximum) for mapping because the highest mirror symmetry was detected when the probe size is ~15 nm or greater [112]. A total of 15x5 CBED patterns were recorded in the area. The step distance between two data points is 15 nm. Among the recorded CBED patterns, we detected two distinguishable mirror symmetries. Figures 7.1(b) and 1(d) show the representative CBED patterns with the two different mirror orientations along the $(11\bar{2})_C$ and $(01\bar{1})_C$. The amount of mirror symmetry was then quantified by the γ using the pairs of 2nd order reflections of A/A′ and B/B′ indicated in Figs. 7.1(b) and (d), which gives $\gamma_{(11\bar{2})_C}$ =0.79/ $\gamma_{(01\bar{1})_C}$ =0.44 and $\gamma_{(11\bar{2})_C}$ =0.61/ $\gamma_{(01\bar{1})_C}$ =0.81 for Figs. 7.1(b) and (d), respectively, with the subscript denoting the direction of the quantified mirror. The 1st order reflections show the similar γ values of ~ 0.85 for $\gamma_{(11\bar{2})_C}$ and $\gamma_{(01\bar{1})_C}$. Thus, the 2nd order reflections are more sensitive to distinguish the two different mirror orientations. Overall, the highest mirror symmetry measured in PMN-31%PT achieved ~81% in cross-correlation, which is significantly lower than the amount of symmetry measured from Si (~0.99).

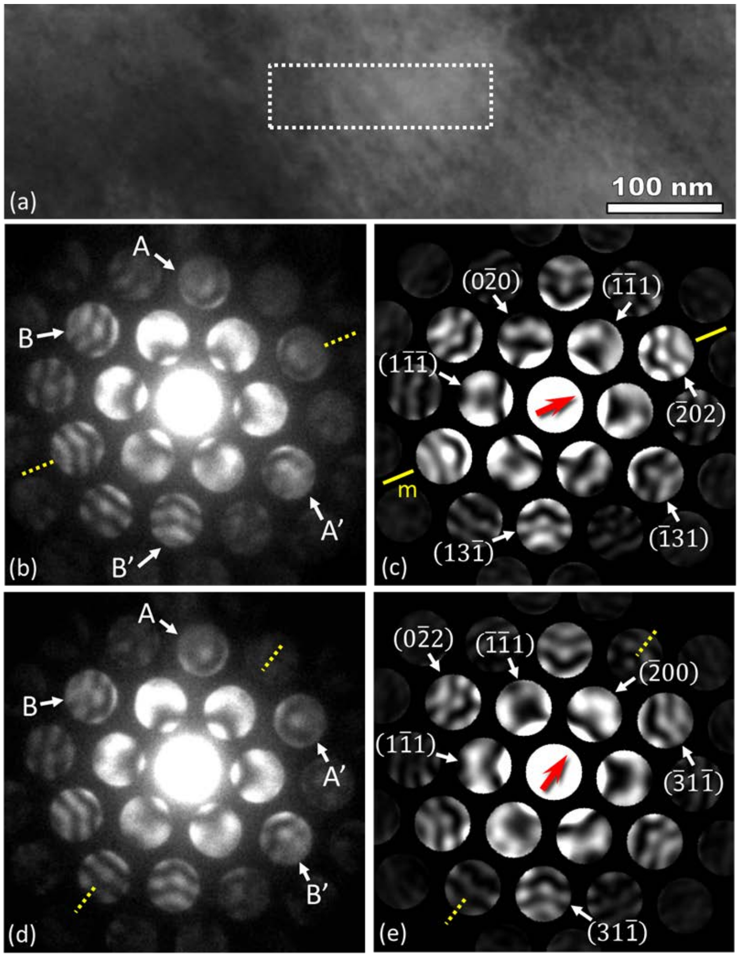


Figure 7.1: (a) A typical BF electron image of PMN-31%PT recorded along $[111]_C$ showing nanoscale tweed-like contrast. The boxed region is scanned for symmetry and polarization mapping. Among the CBED patterns recorded, the (b) and (d) represent CBED patterns with two different types of mirror symmetry. The mirror symmetry is quantified using the pairs of A/A’ and B/B’ indicated in the (b) and (d). The (c) and (e) are the calculated CBED patterns with monoclinic indices for the (b) and (d), respectively.

The two CBED patterns in Figs. 7.1(b) and (d) belong to the zone axes of $[101]_M$ and $[011]_M$, respectively (Ch. 5). The monoclinic distortion induces atomic displacements from the high T cubic symmetry, resulting in a spontaneous polarization. We have estimated the amount of polarization (**P**) based on the best matching structural model. The preferred polarization direction based on the M_B symmetry is along the $[-u0v]_M$ ($u>v$, p=[-1.401, 0, 0.367] Å). Thus, the observed two mirrors indicate polarization domains of two different polarization directions as indicated by the arrow in Figs. 7.1(c) and (e).

Figure 7.2 schematically shows the orientation relationship of these two monoclinic directions in reference to the single crystal pseudo-cubic axes. The monoclinic axes of a_M and b_M are along $<110>_C$ and $c_M \parallel <100>_C$. In the figure, the incident electron beam is taken along $[\bar{1}\bar{1}\bar{1}]_C$. Between the two monoclinic directions, the monoclinic c-axis is rotated by ~90^o with respect to the cubic a-axis. The polarization direction is parallel to the monoclinic mirror plane $(010)_M$ and approximately parallel to $[\bar{1}00]_M$. Considering the small monoclinic distortion, the polarization directions determined by CBED are approximately along $[110]_C$ and $[\bar{1}0\bar{1}]_C$ for Figs. 7.2(b) and (d) respectively with an angle of 60^o. Earlier studies of PMN-xPT have suggested 90^o T or R nanodomains for the observed monoclinic symmetry [54, 58]. The presence of 60^o polarization domains excludes the T or R symmetries since it is only permitted in crystals with orthorhombic or lower symmetries.

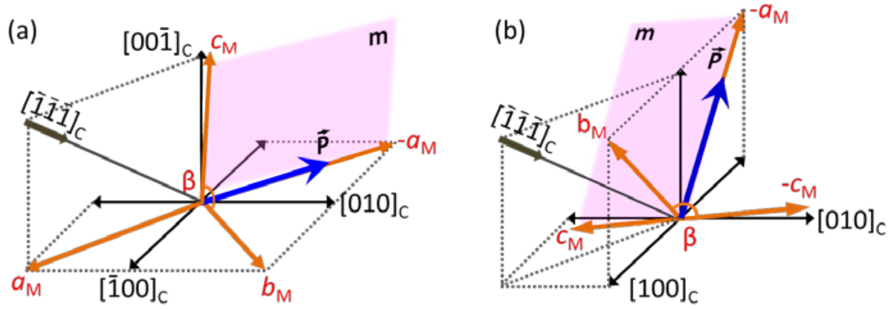


Figure 7.2: Orientation relationship between two nanodomains with respect to the pseudocubic axes. The polarization directions for Figs. 2(b) and (d) correspond to the (a) and (b), respectively.

By combining scanning CBED with the symmetry quantification, we are able to map the symmetry distribution and polarization direction and correlate with the observed image contrast. The results are shown in Fig. 7.3 together with a magnified image of the area investigated by scanning CBED. The tweed-like image contrast is separated into 8 regions and the boundary between these regions is mostly parallel to $(11\bar{2})_C$ as shown in Fig. 7.3(a). The mirror symmetry and polarization direction were quantified by $\gamma_{(\bar{2}02)_M}$ and $\gamma_{(\bar{2}00)_M}$ corresponding to the two mirrors in Figs. 7.1(c) and (e), respectively. A composite symmetry map is constructed and shown in Fig. 7.3(b) using the $\gamma_{(\bar{2}02)_M}$ map for 'red', the $\gamma_{(\bar{2}00)_M}$ map for 'green' (see Sec. S6 in the supplementary) and the electron image intensity in Fig. 7.3(a) for 'blue'. Remarkably, the composite map correlates the observed symmetries and the electron image contrast. The region 'a' is predominately [101]M_B with the $\gamma_{(\bar{2}02)_M}$ values up to 0.79 and the lowest $\gamma_{(\bar{2}00)_M}$ at -0.2, and the region 'c' is dominated by [011]M_B with the highest $\gamma_{(\bar{2}00)_M}$ values of 0.81 and the lowest $\gamma_{(\bar{2}02)_M}$ at 0.45, while the blue regions (labeled as f, g and h in Fig. 7.3) have the low $\gamma_{(\bar{2}02)_M}$ *and* $\gamma_{(\bar{2}00)_M}$ values in the range of -0.21 ~0.37. In the regions 'b' and 'd', the $\gamma_{(\bar{2}02)_M}$ and $\gamma_{(\bar{2}00)_M}$ have similar values of 0.4 ~ 0.6. The regions of 'b' and 'd' are thus considered as transition region between the two domains.

The polarization direction was obtained according to the mirror plane in the CBED patterns. Our above results show that the mirror symmetry is not perfect. To reflect this, we plotted the polarization direction in Fig. 7.3(c) using $\hat{P} = \gamma_{(\bar{2}02)} \overrightarrow{v_{[\bar{1}\bar{1}0]}} + \gamma_{(\bar{2}00)} \overrightarrow{v_{[\bar{1}0\bar{1}]}}$ in short, medium and long arrows scales with magnitude of $\hat{P}$ ranging from 0.51 to 0.57, 0.61 to 0.69 and 0.75 to 1, respectively. The polarization map shown in Fig. 7.3(c) correlates with the electron image (Fig. 7.3(a)) and the composite symmetry map (Fig. 7.3(b)). The polarization map shows the transient regions 'b' and 'd' between predominantly single domains of regions 'a' and 'c'. While the precise width is difficult to measure from the map, the map does suggest a value on the order of electron probe size. In some regions (e to h in Fig. 7.3(c)), the polarization is undetermined. The undetermined region is perhaps a result from strain induced by lattice rotation. Representative CBED patterns given in Fig. 7.3(c) also reflect the polarization rotation shown in the map.

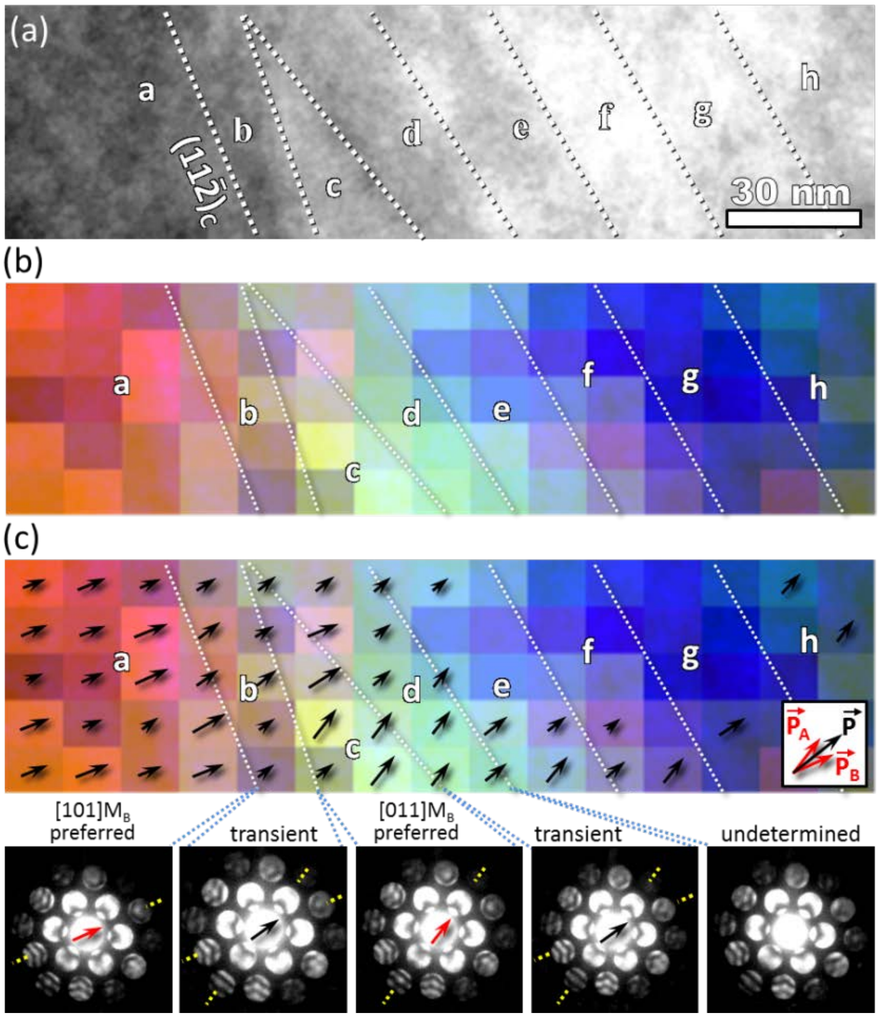


Figure 7.3: (a) A magnified image of the area investigated by scanning CBED. (b) A composite image shows a relation between the symmetry variation and the tweed-like image contrast. The (c) shows a polarization rotation with the representative CBED patterns.

7.3 Summary and conclusion

In summary, the symmetry and polarization direction of nanodomains in PMN-xPT at the MPB composition has been directly determined by CBED. The highest symmetry of monoclinic (M_B) is attained by a ~15 nm electron probe. By combining CBED with scanning, we are able to map the symmetry, polarization direction and lattice direction and correlate these quantities with the tweed-like electron image contrast. The mapping results revealed significant amount of symmetry variations and polarization rotation, including 60° polarization domains and transient regions.

CHAPTER 8

CONCLUSION

In this study, new techniques for symmetry quantification, symmetry mapping and polarization mapping have been developed based on the scanning CBED in order to study complex nanostructure in the relaxor-ferroelectric single crystal. Using the newly developed techniques, symmetry fluctuation is shows as an important crystallographic parameter apart from the average symmetry in complex ferroelectric crystals. Symmetry fluctuation can be measured using CBED. By applying our methods to the unpoled PMN-31%PT single crystal, we demonstrated two aspects of symmetry fluctuation: (1) volume-dependent and (2) spatial dependent fluctuations. The crystal has macroscopic high symmetry (*Cm*-like) only when averaged over a larger volume. The volume averaged symmetry appears to be a linear function of the averaged volume. The averaged symmetry spatially fluctuates at the local volumes of few tens nm^3. It is therefore the spatial symmetry fluctuation indicates the existence of spatially varying polarization. EDX analysis shows significant chemical fluctuations only in volumes of ~20 nm^3 while symmetry fluctuations persist up to ~64 nm^3. Thus, while chemical fluctuation is expected to lead to symmetry breaking, it alone cannot account for the extent of symmetry fluctuations. The other factor causing the symmetry fluctuations will be discussed in a future study. This study provides direct experimental evidence of the nature of symmetry, local- and averaged-polarization in the unpoled relaxor-ferroelectric PMN-31%PT single crystal.

REFERENCES

[1] J. Curie and P. Curie, Bull. de la Société minéralogique de France **3**, 90 (1880).

[2] C. Brown, R. Kell, R. Taylor, and L. Thomas, Component Parts, IRE Transactions on **9**, 193 (1962).

[3] A. F. Devonshire, Phil. Mag. **40**, 1040 (1949).

[4] Author, *Method of Maintaining Electric Currents of Constant Frequency*, US patent 1472583,October 30, 1923

[5] G. W. Pierce, Proceed. Natl. Acad. Sci. **59**, 81 (1923).

[6] J. W. Horton and W. A. Marrison, Proc. Inst. Radio Eng. **16**, 137 (1928).

[7] B. Jaffe, R. S. Roth, and S. Marzullo, J. Res. Nat. Bur. Stand. **55**, 239 (1955).

[8] B. Jaffe, R. S. Roth, and S. Marzullo, J. Appl. Phys. **25**, 809 (1954).

[9] M. J. Madou, *Fundamentals of Microfabrication: The Science of Miniaturization* 2002).

[10] M. W. Barsoum, *Fundamental of Ceramics* (McGraw-Hill, 1997), p. 668.

[11] B. Jaffe, W. R. Cook, and H. Jaffe, *Piezoelectric Ceramics* (Ceramic Books & Literature, 1989).

[12] D. S. Paik, S. E. Park, S. Wada, S. F. Liu, and T. R. Shrout, J. Appl. Phys. **85**, 1080 (1999).

[13] S. E. Park and W. Hackenberger, Curr. Opin. Solid. St. M. **6**, 11 (2002).

[14] S. E. Park and T. R. Shrout, J. Appl. Phys. **82**, 1804 (1997).

[15] D. Berlincourt and H. Jaffe, Phys. Rev. **111**, 143 (1958).

[16] I. A. Ivan, M. Rakotondrabe, J. Agnus, R. Bourquin, N. Chaillet, P. Lutz, J. C. Poncot, R. Duffait, and O. Bauer, Rev. Adv. Mater. Sci. **24**, 1 (2010).

[17] F. Li, S. Zhang, Z. Xu, X. Wei, J. Luo, and T. R. Shrout, J. Appl. Phys. **108**, 034106 (2010).

[18] P. Groth, Ann. der Phys. **217**, 31 (1870).

[19] V. M. Goldschmidt, Trans. Faraday Soc. **25**, 253 (1929).

[20] H. Ouchi, J. Am. Ceram. Soc. **51**, 16 (1968).

[21] H. Ouchi, K. Nagano, and S. Hayakawa, J. Am. Ceram. Soc. **48**, 630 (1965).

[22] H. Ouchi, M. Nishida, and S. Hayakawa, J. Am. Ceram. Soc. **49**, 577 (1966).

[23] S. W. Choi, T. R. Shrout, S. J. Jang, and A. S. Bhalla, Ferroelectrics **100**, 29 (1989).

[24] B. Noheda, D. E. Cox, G. Shirane, J. Gao, and Z. G. Ye, Phys. Rev. B **66**, 054104 (2002).

[25] B. Noheda, Curr. Opin. Solid St. M. **6**, 27 (2002).

[26] B. Noheda, J. A. Gonzalo, L. E. Cross, R. Guo, S. E. Park, D. E. Cox, and G. Shirane, Phys. Rev. B **61**, 8687 (2000).

[27] X.-h. Du, J. Zheng, U. Belegundu, and K. Uchino, Appl. Phys. Lett. **72**, 2421 (1998).

[28] C. A. Randall, N. Kim, J. P. Kucera, W. Cao, and T. R. Shrout, J. Am. Ceram. Soc. **81**, 677 (1998).

[29] S. Mabud, J. Appl. Crystallogr. **13**, 211 (1980).

[30] W. Cao and L. E. Cross, Phys. Rev. B **47**, 4825 (1993).

[31] S. Mishra, D. Pandey, and A. P. Singh, Appl. Phys. Lett. **69**, 1707 (1996).

[32] K. Kakegawa, O. Matsunaga, T. Kato, and Y. Sasaki, J. Am. Ceram. Soc. **78**, 1071 (1995).

[33] M. R. Soares, A. M. R. Senos, and P. Q. Mantas, J. Eur. Ceram. Soc. **19**, 1865 (1999).

[34] M. R. Soares, A. M. R. Senos, and P. Q. Mantas, J. Eur. Ceram. Soc. **20**, 321 (2000).

[35] G. Helke, A. Schonecker, P. Obenaus, U. Keitel, L. Seffner, T. Scholehwar, and U. Lange, in *Applications of Ferroelectrics, 2000. ISAF 2000. Proceedings of the 2000 12th IEEE International Symposium on*2000), pp. 435.

[36] J. George A. Rossetti, W. Zhang, and A. G. Khachaturyan, Appl. Phys. Lett. **88**, 072912 (2006).

[37] V. A. Isupov, Ferroelectrics **266**, 91 (2002).

[38] V. A. Isupov, Phys. Solid State **43**, 2262 (2001).

[39] J. C. Fernandes, D. A. Hall, M. R. Cockburn, and G. N. Greaves, Nucl. Instrum. Meth. B **97**, 137 (1995).

[40] S. K. Mishra, D. Pandey, and A. P. Singh, Appl. Phys. Lett. **69**, 1707 (1996).

[41] B. Noheda, D. E. Cox, G. Shirane, J. A. Gonzalo, L. E. Cross, and S.-E. Park, Appl. Phys. Lett. **74**, 2059 (1999).

[42] A. I. Kingon and J. B. Clark, J. Am. Ceram. Soc. **66**, 253 (1983).

[43] A. I. Kingon and J. B. Clark, J. Am. Ceram. Soc. **66**, 256 (1983).

[44] J. Y. Li, R. C. Rogan, E. Ustundag, and K. Bhattacharya, Nat. Mater. **4**, 776 (2005).

[45] H. X. Fu and R. E. Cohen, Nature **403**, 281 (2000).

[46] B. Noheda and D. E. Cox, Phase. Transit. **79**, 5 (2006).

[47] B. Noheda, Z. Zhong, D. E. Cox, G. Shirane, S. E. Park, and P. Rehrig, Phys. Rev. B. **65** (2002).

[48] D. Vanderbilt and M. H. Cohen, Phys. Rev. B **63**, 094108 (2001).

[49] Z. G. Ye, B. Noheda, M. Dong, D. Cox, and G. Shirane, Phys. Rev. B **64**, 184114 (2001).

[50] J. S. Forrester, R. O. Piltz, E. H. Kisi, and G. J. McIntyre, J. Phys.Condens. Mat. **13**, L825 (2001).

[51] A. K. Singh and D. Pandey, Phys. Rev. B **67**, 064102 (2003).

[52] A. K. Singh, D. Pandey, and O. Zaharko, Phys. Rev. B **74**, 024101 (2006).

[53] S. P. Singh, A. K. Singh, and D. Pandey, J. Phys. Condens. Mat. **19**, 036217 (2007).

[54] Y. U. Wang, Phys. Rev. B **76**, 024108 (2007).

[55] M. Davis, J. Electroceram. **19**, 23 (2007).

[56] S. Bhattacharyya, J. R. Jinschek, H. Cao, Y. U. Wang, J. Li, and D. Viehland, Appl. Phys. Lett. **92**, 142904 (2008).

[57] Y. M. Jin, Y. U. Wang, A. G. Khachaturyan, J. F. Li, and D. Viehland, J. Appl. Phys. **94**, 3629 (2003).

[58] D. Viehland, J. Appl. Phys. **88**, 4794 (2000).

[59] E. H. Kisi, R. O. Piltz, J. S. Forrester, and C. J. Howard, J. Phys. Condens. Mat. **15**, 3631 (2003).

[60] H. T. Stokes, E. H. Kisi, D. M. Hatch, and C. J. Howard, Acta Crystallogr. B **58**, 934 (2002).

[61] A. G. Khachaturyan, S. M. Shapiro, and S. Semenovskaya, Phys. Rev. B **43**, 10832 (1991).

[62] D. A. Porter and K. E. Easterling, *Phase Transformations in Metals and Alloys* (CRC Press, 1992).

[63] A. G. Khachaturyan, Phil. Mag. **90**, 37 (2010).

[64] H. Wang, J. Zhu, X. W. Zhang, Y. X. Tang, and H. S. Luo, J. Am. Ceram. Soc. **91**, 2382 (2008).

[65] H. Wang, J. Zhu, X. W. Zhang, Y. X. Tang, and H. S. Luo, Appl. Phys. Lett. **92**, 132906 (2008).

[66] G. Xu, H. Luo, H. Xu, and Z. Yin, Phys. Rev. B **64**, 020102 (2001).

[67] J. M. Kiat, Y. Uesu, B. Dkhil, M. Matsuda, C. Malibert, and G. Calvarin, Phys. Rev. B **65**, 064106 (2002).

[68] JEOL, http://www.jeolusa.com/

[69] D. B. Williams and C. B. Carter, *Transmission Electron Microscopy: A Textbook for Materials Science* (Plenum Press, 1996), v. 1.

[70] M. De Graef, *Introduction to Conventional Transmission Electron Microscopy* (Cambridge University Press, 2003).

[71] J. C. H. Spence and J. M. Zuo, *Electron microdiffraction* (Plenum Press, 1992).

[72] J. C. H. Spence, *Experimental high-resolution electron microscopy* (Oxford University Press, 1988).

[73] C. Mory, C. Colliex, and J. M. Cowley, **21**, 171 (1987).

[74] Gatan, http://www.gatan.com/imaging/dig_micrograph.php

[75] Gatan, *Digital Micrograph 2.5: Script language reference*.

[76] D. Mitchell, http://www.dmscripting.com/

[77] FELMI-ZFE, http://portal.tugraz.at/portal/page/portal/felmi/DM-Script

[78] B. H. Toby, Powder Diffr. **21**, 67 (2006).

[79] J. P. Lewis, Vision Interface **95**, 120 (1995).

[80] J. M. Zuo, M. Kim, M. O'Keeffe, and J. C. H. Spence, Nature **401**, 49 (1999).

[81] J. M. Zuo, J. C. H. Spence, and R. Hoier, Phys. Rev. Lett. **62**, 547 (1989).

[82] J. M. Zuo, Mater. T. Jim **39**, 938 (1998).

[83] L. V. Azároff, *Elements of X-Ray Crystallography* (TechBooks, 1990).

[84] B. D. Cullity and S. R. Stock, *Elements of x-ray diffraction* (Prentice Hall, 2001).

[85] B. F. Buxton, J. A. Eades, J. W. Steeds, and G. M. Rackham, Phil. Tans. R. Soc. Lond. A **281**, 171 (1976).

[86] M. Loretto, *Electron Beam Analysis of Materials* (Springer, 1993).

[87] F. Bloch, Z. Physik **52**, 555 (1929).

[88] C. Kittel, *Introduction to Solid State Physics* (John Wiley & Sons, 2004).

[89] N. W. Ashcroft and N. D. Mermin, *Solid state physics* (Saunders College, 1976).

[90] H. Bethe, Ann. Phys. (Leipzig) **87**, 55 (1928).

[91] P. A. Doyle and P. S. Turner, Acta Cryst. **A24**, 390 (1968).

[92] *International Tables for Crystallography Volume C: Mathematical, physical and chemical tables* (Springer, 2004).

[93] G. H. Smith and R. E. Burge, Acta Cryst. **15**, 182 (1962).

[94] A. Weickenmeier and H. Kohl, Acta. Cryst. **A47**, 590 (1991).

[95] D. Rez, P. Rez, and I. Grant, Acta Cryst. **A50**, 481 (1994).

[96] K.-H. Kim, D. A. Payne, and J.-M. Zuo, Appl. Phys. Lett. **97**, 261910 (2010).

[97] W.-L. Chan, K. Zhao, N. Vo, Y. Ashkenazy, D. G. Cahill, and R. S. Averback, Phys. Rev. B **77**, 205405 (2008).

[98] Z. Hang, H. Shen, and F. H. Pollak, J. Appl. Phys. **64**, 3233 (1988).

[99] T. G. Bifano, H. T. Johnson, P. Bierden, and R. K. Mali, J. Microelectromech. S. **11**, 592 (2002).

[100] H. Wang, J. Zhu, N. Lu, A. A. Bokov, Z. G. Ye, and X. W. Zhang, Appl. Phys. Lett. **89**, 042908 (2006).

[101] A. Sehirlioglu, D. A. Payne, and P. Han, J. Appl. Phys. **99** (2006).

[102] Y. M. Park, D.-S. Ko, K.-W. Yi, I. Petrov, and Y.-W. Kim, **107**, 663 (2007).

[103] Á. Barna, B. Pécz, and M. Menyhard, **30**, 267 (1999).

[104] R. Schierholz, H. Fuess, K. Tsuda, Y. Ogata, M. Terauchi, and R. Theissmann, Phys. Rev. B **78**, 024118 (2008).

[105] E. A. McLaughlin, T. Liu, and C. S. Lynch, Acta Mater. **53**, 4001 (2005).

[106] H. Song, Y. Li, K. Zhao, H. Zeng, G. Li, and Q. Yin, J. Mater. Res. **24**, 2173 (2009).

[107] K.-H. Kim and J.-M. Zuo, Ultramicroscopy **124**, 71 (2013).

[108] E. Jansen, W. Schafer, and G. Will, J. Appl. Crystallogr. **27**, 492 (1994).

[109] D. M. Bird and Q. A. King, Acta Cryst. **A46**, 202 (1990).

[110] J. Tao *et al.*, Phys. Rev. Lett. **103** (2009).

[111] J. M. Zuo and J. Tao, in *Scanning transmission electron microscopy imaging and analysis*, edited by S. J. Pennycook, and P. D. Nellist (Springer, New York, 2011), pp. 393.

[112] K.-H. Kim, D. A. Payne, and J.-M. Zuo, Phys. Rev. B **86**, 184113 (2012).

[113] M. K. Durbin, J. C. Hicks, S. E. Park, and T. R. Shrout, J. Appl. Phys. **87**, 8159 (2000).

[114] J. K. Lee, J. Y. Yi, K. S. Hong, S. E. Park, and J. Millan, J. Appl. Phys. **91**, 4474 (2002).

[115] J. Tian, P. Han, and D. A. Payne, IEEE T. Ultrason. Ferr. **54**, 1895 (2007).

[116] J. A. Eades, *Convergent-beam diffraction* (A.R. Liss, 1989).

[117] M. Tanaka, Acta Cryst. **A50**, 261 (1994).

[118] J. P. Morniroli and J. W. Steeds, Ultramicroscopy **45**, 219 (1992).

[119] R. Schierholz and H. Fuess, Phys. Rev. B **84**, 064122 (2011).

[120] O. Noblanc, P. Gaucher, and G. Calvarin, J. Appl. Phys. **79**, 4291 (1996).

[121] H. Ohwa, M. Iwata, H. Orihara, N. Yasuda, and Y. Ishibashi, J. Phys. Soc. Jpn. **70**, 3149 (2001).

[122] V. A. Shuvaeva, A. M. Glazer, and D. Zekria, J. Phys. Condens. Mat. **17**, 5709 (2005).

[123] A. Slodczyk, A. Kania, P. Daniel, and A. Ratuszna, J. Phys. D Appl. Phys. **38**, 2910 (2005).

[124] D. Viehland, M. C. Kim, Z. Xu, and J. F. Li, Appl. Phys. Lett. **67**, 2471 (1995).

[125] Z. Xu, M. C. Kim, J. F. Li, and D. Viehland, Phil. Mag. **74**, 395 (1996).

[126] I. Grinberg and A. M. Rappe, Phys. Rev. B **70**, 220101(R) (2004).

[127] M. Suewattana and D. J. Singh, Phys. Rev. B **73**, 224105 (2006).

[128] D. M. Hatch, H. T. Stokes, and W. Cao, J. Appl. Phys. **94**, 5220 (2003).

[129] T. Egami, Ferroelectrics **222**, 421 (1999).

[130] T. Egami, in *Annual Review of Materials Research* (Annual Reviews, Palo Alto, 2007), pp. 297.

[131] C. A. Randall and A. S. Bhalla, Jpn. J. Appl. Phys. Part 1 **29**, 327 (1990).

[132] W. M. Bian, Y. M. Zhu, and L. Zhou, Acta Metall. Sin. **45**, 673 (2009).

[133] D. I. Woodward, J. Knudsen, and I. M. Reaney, Phys Rev B **72** (2005).

[134] R. Schierholz and H. Fuess, J. Appl. Crystallogr. **45**, 766 (2012).

[135] K.-H. Kim, D. A. Payne, and J.-M. Zuo, Phys. Rev. B **86**, 184113 (2012).

[136] L. E. Cross, Ferroelectrics **76**, 241 (1987).

[137] P. K. Davies and M. A. Akbas, J. Phys. Chem. Solids **61**, 159 (2000).

[138] I. K. Jeong, T. W. Darling, J. K. Lee, T. Proffen, R. H. Heffner, J. S. Park, K. S. Hong, W. Dmowski, and T. Egami, Phys. Rev. Lett. **94**, 147602 (2005).

[139] A. Bosak, D. Chernyshov, and S. Vakhrushev, J. Appl. Crystallogr. **45**, 1309 (2012).

[140] M. A. Krivoglaz, *Diffuse scattering of x-rays and neutrons by fluctuations* (Springer, 1996).

[141] N. Zhang, H. Yokota, A. M. Glazer, and P. A. Thomas, Acta Cryst. **B67**, 461 (2011).

[142] Z. Guo, R. Z. Tai, H. J. Xu, C. Gao, G. Q. Pan, H. S. Luo, and K. Namikawa, Appl. Phys. Lett. **91** (2007).

[143] G. A. Smolenskii, V. A. Isupov, A. I. Agranovskaya, and S. N. Popov, Sov. Phys. Sol. State **2**, 2584 (1961).

APPENDIX A: DM SCRIPT FOR SCANNING ELECTRON DIFFRACTION

```
/*/////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
Scanning electron diffraction script, dialogue version

* Descriptions
- 'EMGet-' or 'EMSet-' functions are used to control TEM.
- This function has been tested for JEOL 2010F, JEOL 2100 Cryo TEM and JEOL 2200FS.
- The script must be first calibrated before scanning electron diffraction.

All rights are reserved to Kyouhyun Kim, Ph.D. Candidate, U of Illinois, 2010
/*/////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

// Variables
TagGroup Binning, Exp                   // Binning and exposure time
TagGroup xNum, yNum                     // Data points along x and y direction
TagGroup xLength, yLength                               // Step length along x and y direction
TagGroup RealNum1, RealNum2, RealNum3, RealNum4, RealNum5, RealNum6, RealNum7, RealNum8, RealNum9, RealNum10
TagGroup Name1, Name2, Name3 // Specimen name, Mode, Mangnification
TagGroup ScanOption, CheckScan //Check scan option deleted on 2011/11/02

// The User Interface Frame creates the actions of the buttons
// Define fucntions of each button when it is pressed
class DialogLibraryTestClass4 : uiframe
{
        // f_Start: Actions when 'Start Button' in 'control box 1' is pressed
        void f_Start ( object self)
        {
                        //Gets experimental conditions
                        String Mode, Mag
                        Number Alpha, SpotSize
                        DLGGetValue (Name2, Mode)
                        DLGGetValue (Name3, Mag)
                        DLGGetValue (RealNum7, Alpha)
                        DLGGetValue (RealNum8, SpotSize)

                        //Gets a specimen name
                        String SpecimenName
                        DLGGetValue (Name1, SpecimenName)

                        //Gets a CCD pixel size
                        Number CCDPixels
                        DLGGetValue (RealNum10, CCDPixels)

                        //Gets a binning value
                        number Binning
                        DLGGetValue (RealNum1, Binning)

                        //Gets an exposure time
                        number ExpTime
                        DLGGetValue (RealNum2, ExpTime)

                        //Gets a data point
                        number XPoints, YPoints
                        DLGGetValue (RealNum3, XPoints)
                        DLGGetValue (RealNum4, YPoints)

                        //Gets a step length
                        number StepL_X, StepL_Y, CorF
                        DLGGetValue (RealNum5, StepL_X)
                        DLGGetValue (RealNum6, StepL_Y)

                        //Stores the setting values in TagBase("LastSettingValues")
                        String TagBase=("LastSettingValues_SED")
                        SetPersistentStringNote(TagBase+"Mode", Mode)
                        SetPersistentStringNote(TagBase+"Mag", Mag)
                        SetPersistentNumberNote(TagBase+"Alpha", Alpha)
                        SetPersistentNumberNote(TagBase+"SpotSize", SpotSize)
                        SetPersistentStringNote(TagBase+"SpecimenName", SpecimenName)
                        SetPersistentNumberNote(TagBase+"Binning", Binning)
                        SetPersistentNumberNote(TagBase+"ExpTime", ExpTime)
                        SetPersistentNumberNote(TagBase+"XPoints", XPoints)
                        SetPersistentNumberNote(TagBase+"YPoints", YPoints)
                        SetPersistentNumberNote(TagBase+"StepL_X", StepL_X)
                        SetPersistentNumberNote(TagBase+"StepL_Y", StepL_Y)
                        SetPersistentNumberNote(TagBase+"CorF", CorF)
```

```
// Recall the calibrated unit
TagBase = ("SED_Calibration")
String Units
GetPersistentStringNote(TagBase + "CalUnits", units)

// Recall the calibrated data
TagBase = ("SED_Calibration")
complexnumber c_Horz_UnitVector, c_Vert_UnitVector
number real_HorzScale, real_VertScale
GetPersistentNumberNote(tagBase + "HorzCal", real_HorzScale)
GetPersistentComplexNumberNote(tagBase + "c_Horz_UnitVector", c_Horz_UnitVector)
GetPersistentNumberNote(tagBase + "VertCal", real_VertScale)
GetPersistentComplexNumberNote(tagBase + "c_Vert_UnitVector", c_Vert_UnitVector)

// Store the initial value of deflector: This needs to move the beam back to the initial position after scanning function.
number TDefX_1, TDefY_1
EMGetBeamShift(TDefX_1, TDefY_1)
TagBase=("InitialDefValue0")
SetPersistentNumberNote(TagBase+"xDefValue0", TDefX_1)
SetPersistentNumberNote(TagBase+"yDefValue0", TDefy_1)

//Centered Scan option: The scanning rectangular area has the center with the initial beam position.
ComplexNumber StartDef
Number DefaultScanOption
DefaultScanOption=ScanOption.DlgGetValue()
If (DefaultScanOption==1)
{
StartDef=complex(TDefX_1, TDefY_1)-c_Vert_UnitVector*(0.5*(Ypoints-1)*StepL_Y*real_VertScale)-c_Horz_UnitVector*(0.5*(Xpoints-
1)*real_HorzScale*StepL_X)
SetPersistentComplexNumberNote(TagBase+"StartDef", StartDef)
}

Else
{
StartDef=complex(TDefX_1, TDefY_1)
SetPersistentComplexNumberNote(TagBase+"StartDef", StartDef)
}
// End

GetPersistentComplexNumberNote(TagBase+"StartDef", StartDef)

// Work Image
Number TotSlice=XPoints*YPoints
Image Img3D :=
IntegerImage(SpecimenName+"_"+XPoints+"by"+YPoints+"pts_"+StepL_X+"by"+StepL_Y+units+"_"+Mode+"_"+Mag+"_Alpha("+Alpha+")"+"_Spot("+SpotSize+")",
2, 1, CCDPixels/Binning , CCDPixels/Binning , TotSlice ) // IntegerImage ("Image name", # bytes, singed or not, pixels, pixels, total slice numbers)
Img3D.ShowImage()
Imagedisplay  ImgDisp3D = Img3D.ImageGetImageDisplay(0)
Number Slice_n=0

// Processing time measurement
Number Start=GetHighResTickCount()

// Shift a beam and record an image
Number iY=0
ComplexNumber ScanningDef
while (iy<YPoints)
        {
        number ix=0
        while (ix<XPoints)
                {
        ScanningDef=StartDef+c_Horz_UnitVector*(ix*real_HorzScale*StepL_X)+c_Vert_UnitVector*(iy*StepL_Y*real_VertScale)
                EMSetBeamShift(Real(ScanningDef), Imaginary(ScanningDef))
                Image WorkImage := slice2( Img3D,0,0,Slice_n,0,CCDPixels/Binning,1,1,CCDPixels/Binning,1)
       SSCGainNormalizedBinnedAcquireInPlace (WorkImage , ExpTime, Binning, 0,  0, CCDPixels/Binning, CCDPixels/Binning)
                ImgDisp3D.ImageDisplaySetDisplayedLayers( Slice_n, Slice_n )
                Img3D.UpDateImage()
                OpenAndSetProgressWindow("Current Frame: "+Slice_n,"Total: "+TotSlice,"")
                ix++
                Slice_n++
                }
        ScanningDef=StartDef
        iy++
        }

//Time Measurement
Number End=GetHighResTickCount()
number time_sec = CalcHighResSecondsBetween(start, end)
number time_min = time_sec / 60
Result("Experimental Results\n")
Result("Total Slices: "+TotSlice+"\n")
```

```
        Result("Exposure time: "+ExpTime+"\n")
        Result("Total recording time: " + time_sec.format( "%.2g" ) + " sec. (" + time_min.format( "%.1f") + " min.)\n" )
        Number SliceT=time_sec/TotSlice
        Result("Processing time for each slice: "+SliceT+"\n")

        //Image Calibration
        Img3D.ImageCopyCalibrationFrom(SSCUnprocessedBinnedAcquire(0,Binning,0,0,32,32))
        Img3D.ImageSetDimensionUnitString( 2, "frame" )
        Img3D.ImageSetDimensionScale( 2, 1 )

        // Move the beam back to the initial position
        TagBase=("InitialDefValue0")
        GetPersistentNumberNote(TagBase+"xDefValue0", TDefX_1)
        GetPersistentNumberNote(TagBase+"yDefValue0", TDefY_1)
        EMSetBeamShift(TDefX_1, TDefY_1)
        OKDialog ("Acquition is done.")
    }

    // Creates actions of the buttons in Control Box 2
    //1. Calibration button
    void f_cal(object self) // Calibrate the instrument when 'Calibration button' is pressed.
    {

        IF(!OkCancelDialog("Start 'Search Mode' to calibrate TEM.\nIf TEM is not 'Search Mode', press 'Cancel' and run this program again!"))
        {
            exit(0)
        }

        IF(!OkCancelDialog("1. Please check not to run this mode under 'Diffraction Mode' (200kx GIF).\n2. Choose a proper sample area not
to demage the CCD camera."))
        {
            exit(0)
        }

        image img := GetFrontImage()
        imagedisplay imgDisplay = img.ImageGetImageDisplay(0)

        // Gets image information
        Number CCDPixels
        DLGGetValue (RealNum10, CCDPixels)
        number height = img.ImageGetDimensionSize( 0 )
        number width  = img.ImageGetDimensionSize( 0 )
        number bin = CCDPixels/height
        number scale = img.ImageGetDimensionScale(0)
        string units = img.ImageGetDimensionUnitString(0)

        //Creates ROI boxes on the front image
        imagedisplay imgdisp = img.ImageGetImageDisplay(0)
        ROI ROI1=NewRoi()
        number top, left, bottom, right
        top=height*0.2
        left=width*0.2
        bottom=height*0.8
        right=width*0.8
        ROI1.ROISetRectangle(top, left, bottom, right)
        ROI1.ROISetVolatile(0)
        ROI1.ROISetMoveable(0)
        imgDisplay.ImageDisplayAddROI( ROI1 )

        //Calculate a real scale from the image
        number r_width=abs((bottom-top)*scale)
        number r_height=abs((left-right)*scale)
        result ("width="+r_width+" "+units+","+"r_height="+r_height+" "+units+"\n")

        //Shows ROIs for beam position
        //Beam box 1
        ROI ROI2=NewROI()
        number ROI2_top=top-50/bin
        number ROI2_left=left-50/bin
        number ROI2_bottom=top+50/bin
        number ROI2_right=left+50/bin
        ROI2.ROISetRectangle (ROI2_top, ROI2_left, ROI2_bottom, ROI2_right)
        ROI2.ROISetVolatile(0)
        ROI2.ROISetMoveable(0)
        ROI2.ROISetColor(0,0,1)

        String Label_ROI2="Locate a beam center on this box"
        ROI2.ROISetLabel(label_ROI2)
        imgDisplay.ImageDisplayAddROI( ROI2 )

        //Get deflector's current - 1
```

```
number DefX_1, DefY_1
if(!OKCancelDialog("Use a beam shift knob to locate in the blue box.\nPress 'OK' when you are done."))
                exit(0)
EMGetBeamShift(DefX_1, DefY_1) //FasTEM_GetDefValue (Deflector number, xDefValue, yDefValue)
OKDialog ("Click to continue")
imgDisplay.ImageDisplayDeleteROI(ROI2)

//Beam box 2
ROI2_top=top+50/bin
ROI2_left=right-50/bin
ROI2_bottom=top-50/bin
ROI2_right=right+50/bin
ROI2.ROISetRectangle (ROI2_top, ROI2_left, ROI2_bottom, ROI2_right)
ROI2.ROISetVolatile(0)
ROI2.ROISetMoveable(0)
ROI2.ROISetColor(0,0,1)

ROI2.ROISetLabel(label_ROI2)
imgDisplay.ImageDisplayAddROI( ROI2 )

//Get deflector's current - 2
number DefX_2, DefY_2
if(!OKCancelDialog("Use a beam shift knob to locate in the blue box.\nPress 'OK' when you are done."))
                exit(0)
EMGetBeamShift(DefX_2, DefY_2)
OKDialog ("Click to continue")

//Beam box 3
ROI2_top=bottom+50/bin
ROI2_left=right-50/bin
ROI2_bottom=bottom-50/bin
ROI2_right=right+50/bin
ROI2.ROISetRectangle (ROI2_top, ROI2_left, ROI2_bottom, ROI2_right)
ROI2.ROISetVolatile(0)
ROI2.ROISetMoveable(0)
ROI2.ROISetColor(0,0,1)

ROI2.ROISetLabel(label_ROI2)
imgDisplay.ImageDisplayAddROI( ROI2 )

//Get deflector's current - 3
number DefX_3, DefY_3
if(!OKCancelDialog("Use a beam shift knob to locate in the blue box.\nPress 'OK' when you are done."))
                exit(0)
EMGetBeamShift(DefX_3, DefY_3)
OKDialog ("Click to continue")
imgDisplay.ImageDisplayDeleteROI(ROI1)
imgDisplay.ImageDisplayDeleteROI(ROI2)

//Print results
result("\n----------------------------------------------------------------------\n")
result("\n                    Calibration results\n")
result("\n----------------------------------------------------------------------\n")

//Convert deflector's current to a real unit
complexnumber c_Horz=(complex(DefX_2, DefY_2)-complex(DefX_1, DefY_1))
number real_HorzScale=sqrt(norm(c_Horz))/r_width // Ratio of 'Beam current change' to 'Real unit' (# current/1 nm)
complexnumber c_Horz_UnitVector=c_Horz/sqrt(norm(c_Horz))
complexnumber c_Vert=(complex(DefX_3, DefY_3)-complex(DefX_2, DefY_2))
number real_VertScale=sqrt(norm(c_Vert))/r_height // Ratio of 'Beam current change' to 'Real unit' (# current/1 nm)
complexnumber c_Vert_UnitVector=c_Vert/sqrt(norm(c_Vert)) // Calculate a vertical unit vector

OKDialog ("Please check calibration by using 'Test' button in Calibration box.")

string Time=GetTime(1),Date=GetDate(1)

//Save the calibration result in script
String tagBase = ("SED_Calibration")
SetPersistentStringNote(TagBase + "CalUnits", units)
SetPersistentNumberNote(tagBase + "HorzCal", real_HorzScale)
SetPersistentComplexNumberNote(tagBase + "c_Horz_UnitVector", c_Horz_UnitVector)
SetPersistentNumberNote(tagBase + "VertCal", real_VertScale)
SetPersistentComplexNumberNote(tagBase + "c_Vert_UnitVector", c_Vert_UnitVector)
SetPersistentStringNote(tagBase + "CalUnits", units )
SetPersistentStringNote(tagBase + "CalDate", Date )
SetPersistentStringNote(tagBase + "CalTime", Time )
GetPersistentStringNote(TagBase + "CalDate", Date )
GetPersistentStringNote(TagBase + "CalTime", Time )
result("\n*Last Calibration: "+Date+" "+Time+"\n")

//For check
```

```
        result("\n*Calibrated area\n")
        result ("Width= "+r_width+" "+units+","+"\nHeight= "+r_height+" "+units+"\n")
        result("\nHorizontal Scale= "+real_HorzScale+" (# current/1 )"+units+"\n")
        result("Vertical unit vector= "+c_Horz_UnitVector+" (current)\n")
        result("\nVertical Scale= "+real_VertScale+" (# current/1 )"+units+"\n")
        result("Horizontal unit vector= "+c_Vert_UnitVector+" (current)\n")
        }
        // End of scripts for 'Calibration button' in Control Box 2

}

// Creates the buttons, fields and labels in the dialog
//Script Information
taggroup ScriptInfo()
    {
        taggroup ScriptInfo=dlgcreatelabel("\nScanning Electron Diffraction\n")
        ScriptInfo.dlgexternalpadding(10,0)
        return ScriptInfo
    }
//End


// Creates the buttons in Control Box 1
TagGroup ExperimentalCondition()
{
    TagGroup Box0_Items
    TagGroup Box0=DlgCreateBox("  Experimental Conditions  ", box0_items)

    //Recall the last setting values
    String TagBase=("LastSettingValues_SED")
    String Mode, Mag
    Number Alpha, SpotSize
    GetPersistentStringNote(TagBase+"Mode", Mode)
    GetPersistentStringNote(TagBase+"Mag", Mag)
    GetPersistentNumberNote(TagBase+"Alpha", Alpha)
    GetPersistentNumberNote(TagBase+"SpotSize", SpotSize)

    String Name
    TagGroup Label11=DLGCreateLabel("Mode: ")
    Label11.DLGExternalPadding(0, 0)
    Name2=DLGCreateStringField (Name, 8)
    Name2.DLGExternalPadding(0, 0)
    DLGValue(Name2, Mode)

    TagGroup Label12=DLGCreateLabel("  Mag: ")
    Label12.DLGExternalPadding(0,0)
    Name3=DLGCreateStringField (Name, 10)
    Name3.DLGExternalPadding(0,0)
    DLGValue(Name3, Mag)

    TagGroup Group11=DLGGroupItems(Label11, Name2, Label12, Name3)
    Group11.DLGTableLayout(4, 1, 0)
    Box0_items.DLGAddElement(Group11)
    Group11.DLGExternalPadding(12,3)

    TagGroup Label14=DLGCreateLabel(" Spot Size : ")
    Label14.DLGExternalPadding(0, 0)
    RealNum8=DLGCreateRealField (0,7, 0)
    RealNum8.DLGExternalPadding(0,0)
    DLGValue(RealNum8, SpotSize)

    TagGroup Label13=DLGCreateLabel("Alpha : ")
    Label13.DLGExternalPadding(1,1)
    RealNum7=DLGCreateRealField (0, 7, 0)
    Name2.DLGExternalPadding(0,0)
    DLGValue(RealNum7, Alpha)

    TagGroup Group12=DLGGroupItems(Label14, RealNum8, Label13, RealNum7)
    Group12.DLGTableLayout(4, 1, 0)
    Box0_items.DLGAddElement(Group12)
    Group12.DLGExternalPadding(19, 0)

    Box0.DLGExternalPadding(0, 0)
    Return Box0
}

// Creates the buttons in Control Box 1
TagGroup MainControlBox1()
{
    // Make 'other buttons' - Group 2 starts from here
    taggroup box2_items         //Control box 2 is grouped by 'box2_items//
```

```
taggroup box2=dlgcreatebox("  Control Box 1  ", box2_items)           // Label of 'Control Box 1'

//Recall the last setting values
String TagBase=("LastSettingValues_SED")
number CCDPixels, Binning, ExpTime, Xpoints, YPoints, StepL_X, StepL_Y, CorF
String SpecimenName
GetPersistentStringNote(TagBase+"SpecimenName", SpecimenName)
GetPersistentNumberNote(TagBase+"CCDPixels", CCDPixels)
GetPersistentNumberNote(TagBase+"Binning", Binning)
GetPersistentNumberNote(TagBase+"ExpTime", ExpTime)
GetPersistentNumberNote(TagBase+"XPoints", XPoints)
GetPersistentNumberNote(TagBase+"YPoints", YPoints)
GetPersistentNumberNote(TagBase+"StepL_X", StepL_X)
GetPersistentNumberNote(TagBase+"StepL_Y", StepL_Y)
GetPersistentNumberNote(TagBase+"CorF", CorF)

// Specimen Name
String Name
TagGroup Label10=DLGCreateLabel("1. Specimen Name: ")
Label10.DLGExternalPadding(1,1)
Name1=DLGCreateStringField (Name, 25)
Name1.DLGExternalPadding(0,0)
DLGValue(Name1, SpecimenName)

TagGroup Group10=DLGGroupItems(Label10, Name1)
Group10.DLGTableLayout(2,1,0)
Box2_Items.DLGAddElement(Group10)
Group10.DLGExternalPadding(0,3)

//CCD pixel size: 2010F(1024), 2100 Cryo (2048)
TagGroup CCDLabel=DLGCreateLabel("2. CCD pixel size: ")
CCDLabel.DLGExternalPadding(1,1)
RealNum10=DLGCreateRealField(0,9,0)
RealNum10.DLGExternalPadding(0,0)
DLGValue(RealNum10,CCDPixels)
TagGroup CCDLabel2=DLGCreateLabel("                          ")
CCDLabel2.DLGExternalPadding(1,1)

TagGroup CCDGroup=DLGGroupItems(CCDLabel, RealNum10, CCDLabel2)
CCDGroup.DLGTableLayout(3,1,0)
Box2_Items.DLGAddElement(CCDGroup)
CCDGroup.DLGExternalPadding(0,3)

//Binning and Exposure time
TagGroup label1 = DLGCreateLabel("3. Binning:")
label1.dlgexternalpadding(1,1)
RealNum1 = DLGCreateRealField(0, 7, 2)
RealNum1.dlgexternalpadding(0,0)
DLGValue(RealNum1,Binning)

TagGroup label2 = DLGCreateLabel("   4. Exp time:")
label2.dlgexternalpadding(1,1)
RealNum2 = DLGCreaterealField(0,7,2)
RealNum2.dlgexternalpadding(0,0)
DLGValue(RealNum2,ExpTime)

TagGroup group1 = DLGGroupItems(label1, RealNum1,label2, RealNum2)
group1.DLGTableLayout(6, 1, 0)
box2_items.DLGAddElement(group1)
group1.DLGExternalPadding(0,3)


//2nd row - Step numbers and length
TagGroup label3=DLGCreateLabel("4. Data points      "+"x:")
label3.dlgexternalpadding(1,1)
RealNum3 = DLGCreateRealField(0, 7, 0)
RealNum3.dlgexternalpadding(0,0)
DLGValue(RealNum3,XPoints)

TagGroup label4 = DLGCreateLabel("   y:")
label4.dlgexternalpadding(1,1)
RealNum4 = DLGCreaterealField(0, 7,0)
RealNum4.dlgexternalpadding(0,0)
dlgvalue(RealNum4,YPoints)

TagGroup group2 = DLGGroupItems(label3, RealNum3, label4, RealNum4)
group2.DLGTableLayout(7, 1, 0)
box2_items.dlgaddelement(group2)
group2.dlgexternalpadding(0,3)

//3rd row - Step length
```

```
    TagBase = ("SED_Calibration")
    String Units
    GetPersistentStringNote(TagBase + "CalUnits", units)
    TagGroup label5=DLGCreateLabel("6. Step Length ("+units+")   "+"x:")
    label5.dlgexternalpadding(1,1)
    RealNum5 = DLGCreateRealField(0,7,2)
    RealNum5.dlgexternalpadding(0,0)
    dlgvalue(RealNum5,StepL_X)

    TagGroup label6 = DLGCreateLabel("    y:")
    label6.dlgexternalpadding(1,1)
    RealNum6 = DLGCreaterealField(0,7,2)
    RealNum6.dlgexternalpadding(0,0)
    DLGValue(RealNum6,StepL_Y)

    TagGroup group3 = DLGGroupItems(label5, RealNum5, label6, RealNum6)
    group3.DLGTableLayout(4, 1, 0);
    box2_items.dlgaddelement(group3)
    group3.dlgexternalpadding(0,3)

    //Scan options
    Number defaultscan=0
    ScanOption=DLGCreateCheckBox("Check this box to set an initial position",defaultscan)
    TagGroup ScanLabel=DLGCreateLabel("   as the center of scanning area.        ")
    ScanLabel.DLGExternalPadding(1,1)

    /*
    Number defaultcheckscan=0
    CheckScan=DLGCreateCheckBox("Check Scan",defaultcheckscan)
    */

    /*
    TagGroup ScanOptionsGroup=DLGGroupItems(ScanOption, CheckScan)
    ScanOptionsGroup.DLGTableLayout(2,1,0)
    box2_items.dlgaddelement(ScanOptionsGroup)
    ScanOptionsGroup.dlgexternalpadding(0,3)
    */

    TagGroup ScanOptionsGroup=DLGGroupItems(ScanOption, ScanLabel)
    ScanOptionsGroup.DLGTableLayout(1,2,0)
    box2_items.dlgaddelement(ScanOptionsGroup)
    ScanOptionsGroup.dlgexternalpadding(0,3)

    //Start button
    TagGroup StartButton = DLGCreatePushButton("     Start     ", "f_Start").DLGSide("Center")
    StartButton.dlgexternalpadding(5,5)
    Box2_Items.DlGAddElement(StartButton)

    Box2.DLGExternalPadding(10, 0)
    Return Box2
}


// Creates the buttons in Control Box 2
// Control Box 3 starts here
TagGroup MainControlBox2()
{
    taggroup box3_items          //Control box 3 is grouped by 'box3_items//
    taggroup box3=dlgcreatebox("  Control Box 2  ", box3_items)           // Label of 'Control Box 2'

    String tagBase = ("SED_Calibration")
    String Units
    GetPersistentStringNote(TagBase + "CalUnits", units)
    TagGroup label7 = DLGCreateLabel("Currently Calibrated Units: "+units)
    box3_items.dlgaddelement(label7)

    //Calibration button
    TagGroup CalButton = DLGCreatePushButton("  Calibration  ", "f_Cal").DLGSide("Center")
    CalButton.dlgexternalpadding(72,5)
    box3_items.dlgaddelement(CalButton)
    Box3.DLGExternalPadding(10,0)
    return box3
}
// End of control box 3


//My information
taggroup MyInfo()
    {
            taggroup MyInfo=dlgcreatelabel("Kyouhyun Kim, MatSE, U of Illinois")
            MyInfo.dlgexternalpadding(5,5)
```

```
          return MyInfo
    }
//End



// Function to assemble the compenents of the dialog into the dialog frame
void DialogLibraryScanningED()
{
    // Configure the positioning in the top right of the application window

    TagGroup position;
    position = DLGBuildPositionFromApplication()
    position.TagGroupSetTagAsTagGroup( "Width", DLGBuildAutoSize() )
    position.TagGroupSetTagAsTagGroup( "Height", DLGBuildAutoSize() )
    position.TagGroupSetTagAsTagGroup( "X", DLGBuildRelativePosition( "Inside", 1) )
    position.TagGroupSetTagAsTagGroup( "Y", DLGBuildRelativePosition( "Inside", -1.0 ) )

    TagGroup dialog_items;
    TagGroup dialog = DLGCreateDialog("SED", dialog_items).dlgposition(position);


    // Call each button
    Dialog_Items.DLGAddElement(ScriptInfo())
    Dialog_Items.DLGAddElement(ExperimentalCondition())
   dialog_items.DLGAddElement(MainControlBox1())
   dialog_items.DLGAddElement(MainControlBox2())
   dialog_items.DLGAddElement(MyInfo())


    object dialog_frame = alloc(DialogLibraryTestClass4).init(dialog)
    dialog_frame.display("Scanning Electron Diffraction");
}


// Call the main function which creates and displays the dialog
DialogLibraryScanningED()
```

APPENDIX B: DM SCRIPT FOR SYMMETRY QUANTIFICATION

```
/*/////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

Instruction
(1) Select CBED disks using oval annotation. Multiple sets can be selected for the symmetry quantification.
(2) Cut-off range determines a mask size for calculation.
(3) The selected ROIs (region of interest) can be aligned based on the R-factor and the cross-correlation coefficient. For the alignment, uses a proper step
size. Large step sizes increase the calculation time.
(4) For a single image, the symmetry quantification is calculated using 'Run' button in '2. Quantification for a disp image' menu.
(5) The symmetry mapping is calculated using 'Run' button in '3. Quantification for multislices' menu.

All rights are reserved to Kyouhyun Kim, Ph.D candidte, U of Illinois (2012)
/*/////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////


//************************************************************************
//Subfunctions for the script
//Calcaulate CenterPos & Radius of ROI
void ROIInfo(number TopX, number TopY, number BotX, Number &CenX, Number &CenY, Number &Radius)
{
    Radius=abs(TopX-BotX)*0.5
    CenX=TopX+Radius
    CenY=TopY+Radius
}

//[3] Rotation angle calculation
//aX, aY: center position of ROI A
//bX, bY: center position of ROI B
number Deg2Rad(number Degree)
{
    Return (pi()/180)*Degree
}

number Rad2Ang(number Radian)
{
    Return (180/pi())*Radian
}

number slope(number aX, number aY, number bX, number bY)
{
    Number Angle
    If(aY==bY)
    {
        Angle=0
    }
    If(aX==bX)
    {
        Angle=Deg2Rad(90)
    }
    ELSE
    {
        Number Slope=(bX-AX)/(bY-aY)
        If(Slope>0)
        {
            Angle=Rad2Ang(atan(slope))-90
        }
        If(Slope<0)
        {
            Angle=Rad2Ang(atan(slope))+90
        }
    Angle=-Deg2Rad(Angle)
    }
    return Angle
}

//[4] Mask creation
Image Mask(image Template, number Bot, number Top, number xSize, number ySize, number Correction)
{
    Image Template:=RealImage("", 4, xSize, ySize)
    Number Radius=abs((Bot-Top))*0.5
    Template=tert(iradius<Radius*Correction, 1, 0)
    return Template
}

//[5] ImageRotation
Image Rotation(number ImgID, number TopY, number TopX, number BotY, number BotX, number rotation)
{
    Image Temporary:=GetImageFromID(ImgID)[TopY, TopX, BotY, BotX]
    Image TempR_A:=Rotate(Temporary, Rotation)
    ScrapCopy(TempR_A)
    Number Width, Height
    Number Radius=abs(BotX-TopX)*0.5
    Get2Dsize(TempR_A, Width, Height)
    Image TempR_B:=RealImage("", 4, Width, Height)
    ScrapPaste(TempR_B)
    ScrapMerge(TempR_B)
    return TempR_B
```

```
}


// mean and stdv calculation //
void Statistic(image ref, Image Mask, Number Pixels, Number &mean, Number &STDV)
{
        Number Integ=sum(ref)
        MEAN=Integ/Pixels
        Number Deviation=sum(((ref-mean)*Mask)**2)/Pixels
        STDV=sqrt(Deviation)
}

// r-factor
Number f_Rfactor(Image ImgA, Image ImgB, Image Mask, number xSize, number ySize)
{
        Number numerator, denominator
        Numerator=sum((ImgB-ImgA)**2)
        Denominator=sum(ImgA*ImgA)
        return sqrt(numerator/denominator)
}

// Cross-correlation coefficient calculation
Number XCorr(Image ImgA, Image ImgB, Image Mask, Number Pixels)
{
 Number Numer, Denom
 Number ImgAMean=sum(ImgA)/pixels
 Number ImgBMean=sum(ImgB)/pixels
 Numer=sum(((ImgA-ImgAMean)*(ImgB-ImgBMean))*Mask)
 Denom=sum(((ImgA-ImgAMean)*Mask)**2)*sum(((ImgB-ImgBMean)*Mask)**2)
 Denom=sqrt(Denom)
 return Numer/Denom
}

// Crop Image function
Image ImageCrop(Image ResultImage, Number ROIRadius, number CutOff)
{
        Image Temp
        Number Width, Height, Coeff=0
        Get2DSize(ResultImage, Width, Height)
        ScrapCopy(ResultImage[Height*0.5-ROIRadius, Width*0.5-ROIRadius, Height*0.5+ROIRadius+Coeff, Width*0.5+ROIRadius+Coeff])
        Temp:=RealImage("", 4, 2*ROIRadius+Coeff, 2*ROIRadius+Coeff)
        ScrapPaste(Temp)
        ScrapMerge(Temp)
        return Temp
}

//[10] Set Background color as white
Image ImageAdj(Image ResultImage, number MaxValue, Number ROIRadius, number CutOff)
{
        Image Temp=ResultImage
        ROIRadius=ROIRadius*CutOff
        If (MaxValue<=1)
        {
        SetLimits(Temp, 0, 1)
        Temp=Tert(iRadius>ROIRadius, 1, Temp)
        }
        ELSE IF (MaxValue>1)
        {
        Temp=Tert(iRadius>ROIRadius, MaxValue, Temp)
        }
        Return Temp
}

//[11] Max, Min value calculation
void MaxMin(Image Source, Number &MaxVal, Number &xMax, Number &yMax, Number &MinVal, Number &xMin, Number &yMin)
{
 MaxVal=max(Source, xMax, yMax)
 MinVal=min(Source, xMin, yMin)
}

//End of Sub_functions
//******************************************************************************

//---------------------------------------------------------------------------
// Variables I
Number ImageID, CutOff
Number AnnotA, AnnotB //Annotation ID
Number A_AnnTx, A_AnnTy, A_AnnBx, A_AnnBy
Number B_AnnTx, B_AnnTy, B_AnnBx, B_AnnBy
Number RadiusA, RadiusB
Number xWidth, yHeight
Number A_CenterX, A_CenterY, B_CenterX, B_CenterY
Number RotAngle
Number NoOfPixels
Number mean_A, stdv_A
Number xPix, yPix, Slices // to handle mutislices xPix, yPix=physical size of image
Number rFactor, CrossCorrelationCoeff, SumSquaredDifference

// Variables II
Number xMove, yMove
Number xStep, yStep

Number NoOfAnnot
```

```
Number TypeOfAnnot
Image AnnotIDSave=RealImage("", 4, 1, 40)
Image AnnotIDSave2=RealImage("", 4, 1, 40)
Number Annot1_ID=GetImageID(AnnotIDSave)
Number Annot2_ID=GetImageID(AnnotIDSave2)

Image MatCoord1=RealImage("", 4, 5, 40)
Image MatCoord2=RealImage("", 4, 5, 40)
Image MatCoord3=RealImage("", 4, 5, 40)
Image MatCoord4=RealImage("", 4, 5, 40)
Number Coord1_ID=GetImageID(MatCoord1)
Number Coord2_ID=GetImageID(MatCoord2)
Number Coord3_ID=GetImageID(MatCoord3)
Number Coord4_ID=GetImageID(MatCoord4)

Number TempAnnotID, Row, Row2, SetsCount, Sets, SetsCountB

//-----------------------------------------------------------------------------


//---------------------------------------------------------------------------
//Variables for DLG
TagGroup Value1, Value2, Value3, Value4, LegendOption, AdjIntRatio, AdjXCorr, Value5, Value6, Value7, Value8, AdjRfactor
TagGroup TwoOption, ThreeOption, FourOption, SixOption, MirrorOption, TemplateOption, IntDiffOption1, IntDiffOption2
TagGroup Coord1, Coord2, Coord3, Coord4, Coord1B, Coord2B, Coord3B, Coord4B
Number DefaultCutOff=0.9
Number DefaultLegend=0
Number DefaultXCorr=1
Number DefaultIntRatio=0
Number DefaultRfactor=0
Number DefaultTemplate=0
Number DefaultIntDiff1=0
Number DefaultIntDiff2=0
Number DefaultTwo=0
Number DefaultThree=0
Number DefaultFour=0
Number DefaultSix=0
Number DefaultMirror=1
Number DefaultCoord1=0, DefaultCoord2=0, DefaultCoord3=0, DefaultCoord4=0
Number DefaultCoord1B=0, DefaultCoord2B=0, DefaultCoord3B=0, DefaultCoord4B=0

Number xPoints, yPoints
Number xCor, yCor
//---------------------------------------------------------------------------

// User interface: Define the actions of the buttons************************************************************
class DialogLibraryTestClass4 : uiframe
{
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//                                                            ////////////////////////////////////////////////////////
//   Automatic alignment                                      ////////////////////////////////////////////////////////
//                                                            ////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
void f_AutoFind(object self)
  {
    DLGGetValue(Value1, xMove)
    DLGGetValue(Value2, xStep)
    DLGGetValue(Value3, yMove)
    DLGGetValue(Value4, yStep)
            Self.DLGGetValue("CutOffRange", CutOff)

            Result("\n------------------------------------------------------------------------\n")
    Result("\n                  CBED symmetry quantification                    \n")
    Result("        Kyouhyun Kim, Ph.D candidate, MatSE, U of Illinois ")
    Result("\n                          2012                       \n")
    Result("\n------------------------------------------------------------------------\n")
    Result("\n*ROI's poisition adjusted")
    Result("\n\n*Selected options*")
            Result("\n(1) Cut-off range: "+CutOff+" ("+CutOff+" of ROI's area used for caculation")

            //Initialize matrix for annotaion ID save function
            AnnotIDSave=AnnotIDSave*0
            AnnotIDSave2=AnnotIDSave2*0

            // Scrap displayed image
            Image Front:=GetFrontImage() // Base image for calculation
            ImageDisplay ImgDisp = Front.ImageGetImageDisplay(0)
            GetSize(Front, xPix, yPix)
            Image TempScrap:=RealImage("Temp Scrap", 4, xPix, yPix)
            TempScrap=ImgDisp.ImageDisplayGetDisplayedImage()
            ImageID=GetImageID(TempScrap)

            // Assign annotations' IDs to AnnotA and AnnotB
            Row=0
            SetsCount=0
            Sets=1
            Number i=0
            Number ix=0
    NoOfAnnot=CountAnnotations(Front)

            While (i<NoOfAnnot)
```

```
{
              TempAnnotID=GetNthAnnotationID(Front, i)
              TypeOFAnnot=AnnotationType(Front, TempAnnotID)
              If(TypeOfAnnot==5||TypeOfAnnot==6)
              {
                        If(TypeOfAnnot==6)
                        {
                        AnnotIDSave.SetPixel(0, row, TempAnnotID)
                        row++
                        }
                        If(TypeOfAnnot==5)
                        {
                        AnnotIDSave2.SetPixel(0, row2, TempAnnotID)
                        row2++
                        ix++
                        }
                        i++
                        Sets++
              }
              Else
              {
                        i++
              }
    }
    Sets=(Sets-1)/2
    Result("\n>"+(Sets+ix*0.5)+" set(s) selected")
    Result("\n>"+ix+" discs are placed on the mirror plane")

    DLGGetValue(TwoOption, DefaultTwo)
DLGGetValue(ThreeOption, DefaultThree)
DLGGetValue(FourOption, DefaultFour)
DLGGetValue(SixOption, DefaultSix)
DLGGetValue(MirrorOption, DefaultMirror)

//Auto calculation loop start here
    SetsCountB=0
    Number RotAngleTemp
    Number nyy=0
    While (SetsCount<Sets) //***************************************************************************
    {
    If (SetsCount<(Sets-ix*0.5))
    {
              AnnotA=AnnotIDSave.GetPixel(0, SetsCount*2+1)
              AnnotB=AnnotIDSave.GetPixel(0, SetsCount*2)
              GetAnnotationRect(Front, AnnotA, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
              GetAnnotationRect(Front, AnnotB, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx)
              ROIInfo(A_AnnTx, A_AnnTy, A_AnnBx, A_CenterX, A_CenterY, RadiusA)
              ROIInfo(B_AnnTx, B_AnnTy, B_AnnBx, B_CenterX, B_CenterY, RadiusB)
              If (RadiusA!=RadiusB)
              {
                        B_AnnTx=B_centerX-RadiusA
                        B_AnnTy=B_centery-RadiusA
                        B_AnnBx=B_centerX+RadiusA
                        B_AnnBy=B_centery+RadiusA
              }
              CreateTextAnnotation(Front, A_CenterY, A_CenterX, "A"+(SetsCount+1))
              CreateTextAnnotation(Front, B_CenterY, B_CenterX, "B"+(SetsCount+1))
    }

    If (SetsCount>=(Sets-ix*0.5))
    {
              AnnotA=AnnotIDSave2.GetPixel(0, SetsCountB+1)
              AnnotB=AnnotIDSave2.GetPixel(0, SetsCountB)
              GetAnnotationRect(Front, AnnotA, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
              GetAnnotationRect(Front, AnnotB, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx)
              ROIInfo(A_AnnTx, A_AnnTy, A_AnnBx, A_CenterX, A_CenterY, RadiusA)
              ROIInfo(B_AnnTx, B_AnnTy, B_AnnBx, B_CenterX, B_CenterY, RadiusB)
              If (RadiusA!=RadiusB)
              {
                        B_AnnTx=B_centerX-RadiusA
                        B_AnnTy=B_centery-RadiusA
                        B_AnnBx=B_centerX+RadiusA
                        B_AnnBy=B_centery+RadiusA
              }
              CreateTextAnnotation(Front, A_CenterY, A_CenterX, "m"+(SetsCountB+2))
              CreateTextAnnotation(Front, B_CenterY, B_CenterX, "m"+(SetsCountB+1))
              SetsCountB=SetsCountB+2
    }

If (SetsCount<(Sets-ix*0.5))
{
    Result("\n\n*Set "+(SetsCount+1)+" alignment")
    }
    Else If (SetsCount>=(Sets-ix*0.5))
{
    Result("\n\n*Set m("+(SetsCountB-1)+", "+(SetsCountB)+")"+" alignment")
    }

Number B_InitTy, B_InitTx, B_InitBy, B_InitBx
B_InitTy=B_AnnTy
B_InitTx=B_AnnTx
B_InitBy=B_AnnBy
B_InitBx=B_AnnBx
```

```
Number Loops=1

Number nx=0, ny=0, EndProcess=0
Number CalXPos, CalYPos
Number AdjTop, AdjBot

Image TempAdj:=RealImage("TempAdj", 4, xMove*2, yMove*2)
Image WorkImageA, WorkImageB, WorkRotatedA, WorkRotatedB, Temp_A, Temp_B, ResultA, ResultB, FlipTemp

while(ny<yMove*2)
 {
    B_AnnTy=B_InitTy+(-yMove*yStep)+(ny*yStep)
    B_AnnBy=B_InitBy+(-yMove*yStep)+(ny*yStep)
  B_CenterY=B_AnnTy+abs((B_AnnBy-B_AnnTy)*0.5)
    while(nx<xMove*2)
    {
    B_AnnTx=B_InitTx+(-xMove*xStep)+(nx*xStep)
    B_AnnBx=B_InitBx+(-xMove*xStep)+(nx*xStep)
    B_CenterX=B_AnnTx+abs((B_AnnBx-B_AnnTx)*0.5)

                                //New function starts here**************************************************************8
                                Image Temp_A, Temp_B, TempMask
                                Image TempAAA, TempBBB

                                // Rotation symmetry
                                If (DefaultTwo==1||DefaultThree==1||DefaultFour==1||DefaultSix==1)
                                {
                                        RotAngle=slope(A_CenterX, A_CenterY, B_CenterX, B_CenterY)
                                        RotAngle=0
                                        Image TempSymm_A=Rotation(ImageID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngle)
                                        Temp_A=ImageCrop(TempSymm_A, RadiusA, 1)
                                        Image TempSymm_B=Rotation(ImageID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngle)

                                        If (DefaultTwo==1)
                                        {
                                                RotAngle=Deg2Rad(180)
                                        }
                                        If (DefaultThree==1)
                                        {
                                                RotAngle=Deg2Rad(120)
                                        }
                                                If (DefaultFour==1)
                                        {
                                                RotAngle=Deg2Rad(90)
                                        }
                                        If (DefaultSix==1)
                                        {
                                                RotAngle=Deg2Rad(60)
                                        }
                                        Temp_B:=ImageCrop(Rotate(TempSymm_B, RotAngle), RadiusA, 1)
                                }

                                // Mirror symmetry
                                If (DefaultMirror==1)
                                {
                                        RotAngle=slope(A_CenterX, A_CenterY, B_CenterX, B_CenterY)
                                        Temp_A=Rotation(ImageID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngle)
                                        Temp_B=Rotation(ImageID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngle)
                                }

                                //Create a mask
                                Get2Dsize(Temp_A, xWidth, yHeight)
                                TempMask=Mask(TempMask, A_AnnBx, A_AnnTx, xWidth, yHeight, CutOff)
                                NoOfPixels=sum(TempMask) //Calculate a number of pixes inside the created mask

                                //(4) Mask x Image
                                If (DefaultMirror==1)
                                {
                                        If (SetsCount<(Sets-ix*0.5))
                                        {
                                                FlipHorizontal(Temp_B)
                                                Temp_A=Temp_A*TempMask
                                                Temp_B=Temp_B*TempMask
                                        }
                                        Else If (SetsCount>=(Sets-ix*0.5))
                                        {
                                                TempAAA=Temp_A
                                                FlipVertical(TempAAA)
                                                Temp_A=Temp_A*TempMask
                                                TempAAA=TempAAA*TempMask

                                                TempBBB=Temp_B
                                                FlipVertical(TempBBB)
                                                Temp_B=Temp_B*TempMask
                                                TempBBB=TempBBB*TempMask
                                        }
                                }
                                Else
                                {
                                        Temp_B=Temp_B*TempMask
                                }
```

```
      // ROI adjustment with max normalized cross-correlation coefficient
      AdjXcorr.DLGGetValue(DefaultXCorr)
                                    If(DefaultXCorr==1)
                                    {
                                              If(SetsCount>=(Sets-ix*0.5))
                                              {
                                                        CrossCorrelationCoeff=0.5*(Xcorr(Temp_A, TempAAA, TempMask,
NoOfpixels)+Xcorr(Temp_B, TempBBB, TempMask, NoOfpixels))
                                                        Result("\nXcross="+CrossCorrelationCoeff)
                                              }
                                              Else
                                              {
                                                        CrossCorrelationCoeff=Xcorr(Temp_A, Temp_B, TempMask, NoOfpixels)
                                              }
                                              SetPixel(TempAdj, nx, ny, CrossCorrelationCoeff)
                                    }

      // ROI adjustment with weighted R-factor
      AdjRfactor.DLGGetValue(DefaultRfactor)
                                    If(DefaultRfactor==1)
                                    {
                                              rFactor=f_Rfactor(Temp_A, Temp_B, TempMask, xWidth, yHeight)
                                              SetPixel(TempAdj, nx, ny, rFactor)
                                    }
      nx++
      OpenAndSetProgressWindow("Searching for Set"+(SetsCount+1)+"...",round((Loops/(xMove*yMove*4)*100))+"% completed","")
                                    Loops++
      } //for nx
    nx=0
    ny++
   } //for ny

   AdjRfactor.DLGGetValue(DefaultRfactor)
       If(DefaultRfactor==1)
       {
                min(TempAdj, CalxPos, CalyPos)
                Result("\nMin weighted R-factor between two ROIs ")
                Result("\n> Min="+min(TempAdj))
       }

   AdjXcorr.DLGGetValue(DefaultXCorr)
       If(DefaultXCorr==1)
       {
                max(TempAdj, CalxPos, CalyPos)
                Result("\n> Max="+max(TempAdj))
       }
       Result("\n> adjusted by")
   AdjTop=(-yMove*yStep)+CalyPos*yStep
   AdjBot=(-xMove*xStep)+CalxPos*xStep
   Result(" x= "+AdjTop+" pixels, y= "+AdjBot+" pixels")
   MoveAnnotation(Front, AnnotB, B_InitTy+AdjTop, B_InitTx+AdjBot, B_InitBy+AdjTop, B_InitBx+AdjBot)

      SetsCount++
  }
 }//End of void 'f_AutoFind'

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//                                                            ////////////////////////////////////////////////////////
//   Symmetry quantification for a displayed image          ////////////////////////////////////////////////////////
//                                                            ////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

 void f_SingleImage(object self)
 {
      // Scrap displayed image
      Image Front:=GetFrontImage() // Base image for calculation
      ImageDisplay ImgDisp = Front.ImageGetImageDisplay(0)
      GetSize(Front, xPix, yPix)
      Image TempScrap:=RealImage("Temp Scrap", 4, xPix, yPix)
      TempScrap=ImgDisp.ImageDisplayGetDisplayedImage()
      ImageID=GetImageID(TempScrap)

      Self.DLGGetValue("CutOffRange", CutOff)


      Result("\n-------------------------------------------------------------------------\n")
  Result("\n                CBED symmety quantification                \n")
  Result("      Kyouhyun Kim, Ph.D candidate, MatSE, U of Illinois ")
  Result("\n                   2012                    \n")
  Result("\n-------------------------------------------------------------------------\n")
  Result("\n**Selected options**")
      Result("\n>Cut-off range: "+CutOff+" ("+CutOff+" of ROI's area used for caculation)")


      //Initialize matrix for annotaion ID save function
      AnnotIDSave=AnnotIDSave*0
      AnnotIDSave2=AnnotIDSave2*0

      // Assign annotations' IDs to AnnotA and AnnotB
      Row=0
      Row2=0
```

```
    SetsCount=0
    Sets=1
    Number SetsTemp

    Number i=0
    Number ix=0
NoOfAnnot=CountAnnotations(Front)
    While (i<NoOfAnnot)
{
            TempAnnotID=GetNthAnnotationID(Front, i)
            TypeOFAnnot=AnnotationType(Front, TempAnnotID)
            If(TypeOfAnnot==5||TypeOfAnnot==6)
            {
                    If(TypeOfAnnot==6)
                    {
                    AnnotIDSave.SetPixel(0, row, TempAnnotID)
                    row++
                    }
                    If(TypeOfAnnot==5)
                    {
                    AnnotIDSave2.SetPixel(0, row2, TempAnnotID)
                    row2++
                    ix++
                    }
                    i++
                    Sets++
            }
            Else
            {
                    i++
            }
    }
    Sets=(Sets-1)/2+0.5*ix
    Result("\n>"+Sets+" set(s) selected")
    Result("\n>"+ix+" discs are placed on the mirror plane")

    rFactor=0
    CrossCorrelationCoeff=0

    DLGGetValue(TwoOption, DefaultTwo)
DLGGetValue(ThreeOption, DefaultThree)
DLGGetValue(FourOption, DefaultFour)
DLGGetValue(SixOption, DefaultSix)
DLGGetValue(MirrorOption, DefaultMirror)

    SetsCountB=0
    Number RotAngleTemp
    Number nxx=0, nyy=0
    Result("\n\n            X-corr                  r-Factor")
    While (SetsCount<Sets) //********************************************************************************
    {
    // Allocate annotations' IDs and get annotaions' info

    If (SetsCount<(Sets-ix))
    {
            AnnotA=AnnotIDSave.GetPixel(0, SetsCount*2+1)
            AnnotB=AnnotIDSave.GetPixel(0, SetsCount*2)
            GetAnnotationRect(Front, AnnotA, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
            GetAnnotationRect(Front, AnnotB, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx)

            //Calculate Radius & Rotation Angle
            ROIInfo(A_AnnTx, A_AnnTy, A_AnnBx, A_CenterX, A_CenterY, RadiusA)
            ROIInfo(B_AnnTx, B_AnnTy, B_AnnBx, B_CenterX, B_CenterY, RadiusB)
            If (RadiusA!=RadiusB)
            {
                    B_AnnTx=B_centerX-RadiusA
                    B_AnnTy=B_centery-RadiusA
                    B_AnnBx=B_centerX+RadiusA
                    B_AnnBy=B_centery+RadiusA
            }
            CreateTextAnnotation(Front, A_CenterY, A_CenterX, "A"+(SetsCount+1))
            CreateTextAnnotation(Front, B_CenterY, B_CenterX, "B"+(SetsCount+1))
            RotAngle=slope(A_CenterX, A_CenterY, B_CenterX, B_CenterY)
            /*
            Result ("\n\nRotation angle="+Rad2Ang(RotAngle)+"\n")
            */
    }
    If (SetsCount>=(Sets-ix))
    {
            If (nxx<1) //nx used to calculate rotatiion angle of disc lied on the mirror axis for one time
            {
            AnnotA=AnnotIDSave2.GetPixel(0, SetsCountB+1)
            AnnotB=AnnotIDSave2.GetPixel(0, SetsCountB)
            GetAnnotationRect(Front, AnnotA, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
            GetAnnotationRect(Front, AnnotB, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx)
            ROIInfo(A_AnnTx, A_AnnTy, A_AnnBx, A_CenterX, A_CenterY, RadiusA)
            ROIInfo(B_AnnTx, B_AnnTy, B_AnnBx, B_CenterX, B_CenterY, RadiusB)
            If (RadiusA!=RadiusB)
            {
                    B_AnnTx=B_centerX-RadiusA
                    B_AnnTy=B_centery-RadiusA
                    B_AnnBx=B_centerX+RadiusA
                    B_AnnBy=B_centery+RadiusA
```

```
            }
            CreateTextAnnotation(Front, A_CenterY, A_CenterX, "m"+(SetsCountB+2))
            CreateTextAnnotation(Front, B_CenterY, B_CenterX, "m"+(SetsCountB+1))
            RotAngleTemp=slope(A_CenterX, A_CenterY, B_CenterX, B_CenterY) //Calculate rotation angle first
            SetsCountB=SetsCountB+2
            }
    }
Image Temp_A, Temp_B, TempMask
    Image TempAAA, TempBBB
// Rotation symmetry
    If (DefaultTwo==1||DefaultThree==1||DefaultFour==1||DefaultSix==1)
    {
            RotAngle=0
            Image TempSymm_A=Rotation(ImageID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngle)
            Temp_A=ImageCrop(TempSymm_A, RadiusA, 1)
            Image TempSymm_B=Rotation(ImageID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngle)

            If (DefaultTwo==1)
            {
                    RotAngle=Deg2Rad(180)
            }
            If (DefaultThree==1)
            {
                    RotAngle=Deg2Rad(120)
            }
            If (DefaultFour==1)
            {
                    RotAngle=Deg2Rad(90)
            }
            If (DefaultSix==1)
            {
                    RotAngle=Deg2Rad(60)
            }
            Temp_B:=ImageCrop(Rotate(TempSymm_B, RotAngle), RadiusA, 1)
            Temp_A.SetName("Disk A_Set "+(SetsCount+1))
            Temp_B.SetName("Disk B_Set "+(SetsCount+1))
    }

// Mirror symmetry
    If (DefaultMirror==1)
    {
            //Image rotation
            If (SetsCount>=(Sets-ix))
            {
                    TempAAA=Rotation(ImageID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngleTemp)
                    TempBBB=Rotation(ImageID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngleTemp)
                    If (nyy==0)
                    {
                            Temp_A=TempAAA
                            Temp_B=TempAAA
                            nxx++
                            nyy++
                    }
                    Else If (nyy==1)
                    {
                            Temp_A=TempBBB
                            Temp_B=TempBBB
                            nyy=0
                            nxx=0
                    }
            }
            Else
            {
                    /*
                    Result("\n"+A_AnnTy+"      "+A_AnnTx+""+A_AnnBy+"                    "+A_AnnBx)
                    Result("\n"+B_AnnTy+"      "+B_AnnTx+""+B_AnnBy+"                    "+B_AnnBx)
                    */
                    Temp_A=Rotation(ImageID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngle)
                    Temp_B=Rotation(ImageID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngle)
            }
    }

    //Create a mask
Get2Dsize(Temp_A, xWidth, yHeight)
    TempMask=Mask(TempMask, A_AnnBx, A_AnnTx, xWidth, yHeight, CutOff)
    NoOfPixels=sum(TempMask) //Calculate a number of pixes inside the created mask (to be used for quantification)


    // Generate Mask Image ->  generate mirror image -> apply
    //(4) Mask x Image
    Temp_A=Temp_A*TempMask

    Image Temp_AA, Temp_BB, FlipTemp
    If (DefaultMirror==1)
    {
            If (SetsCount>=(Sets-ix))
            {
                    Temp_A.SetName("aligned m_Set "+(SetsCount+1))
                    FlipVertical(Temp_B)
                    Temp_B=Temp_B*TempMask
                    Temp_B.SetName("mirror applied_Set "+(SetsCount+1))
            }
```

```
            Else
            {
                Temp_A.SetName("aligned A_Set "+(SetsCount+1))
                FlipHorizontal(Temp_B)
                /*
                Temp_A.ShowImage()
                Temp_B.ShowImage()
                TempMask.ShowImage()
                */
                Temp_B=Temp_B*TempMask
                Temp_B.SetName("symm applied B_set "+(SetsCount+1))
                Temp_BB=Temp_B
                FlipHorizontal(Temp_BB)
                Temp_BB.SetName("aligned B_set "+(SetsCount+1))
            }
    }

    Else
    {
            Temp_B=Temp_B*TempMask
            Temp_B.SetName("Disk B_Set "+(SetsCount+1))
    }

    // Show caclculated templates
    TemplateOption.DLGGetValue(DefaultTemplate)
    If(DefaultTemplate==1)
    {
            Temp_A.ShowImage()
            Temp_B.ShowImage()
            If (DefaultMirror==1)
            {
                If (SetsCount<(Sets-ix))
                {
                    Temp_BB.ShowImage()
                }
            }
    }

// (7) R-factor
Number Temp_rFactor=f_Rfactor(Temp_A, Temp_B, TempMask, xWidth, yHeight)

    // Cross-correlation coefficient
    Number Temp_XCross=XCorr(Temp_A, Temp_B, TempMask, NoOfPixels)

    //Return values
    If (SetsCount<(Sets-ix))
    {
    Result("\nSet "+(SetsCount+1)+" "+"       "+Temp_XCross+"             "+Temp_rFactor)
    }
    Else
    {
    Result("\nSet "+"m"+(SetsCount-(Sets-ix)+1)+"           "+Temp_XCross+"             "+Temp_rFactor)
    }
    rFactor=rFactor+Temp_rFactor
    CrossCorrelationCoeff=CrossCorrelationCoeff+Temp_XCross

// (8) Intensity difference ratio (%)
Image ResultB=IntRatio(Temp_A, Temp_B, xWidth, yHeight)
ResultB.SetName("Intensity difference ratio (%)_Set "+(SetsCount+1))

// Intensty difference (A-B)
Image ResultBB
ResultBB=abs(Temp_A-Temp_B)
ResultBB.SetName("A-B_Set "+(SetsCount+1))

    LegendOption.DLGGetValue(DefaultLegend)
    If(DefaultLegend==1)
            {
                Image Legend:=RealImage("Legend", 4, xWidth*0.05, yHeight*0.7)
                Legend=irow
                Legend.SetColorMode(4)
                FlipVertical(Legend)
                ScrapCopy(Legend)
                ScrapPaste(ResultB)
            }
    SetColorMode(ResultB, 4)

    number maxb, xb, yb
    IntDiffOption1.DLGGetValue(DefaultIntDiff1)
    If(DefaultIntDiff1==1)
    {
Result("\n\n> Intensity difference ratio(%)")
Statistic(ResultB, TempMask, NoOfPixels, mean_A, STDV_A)
Result("\nMean: "+mean_A)
Result("\nStandard deviation: "+STDV_A)

maxb=max(resultB,xb, yb)
result("\nMax= "+maxb)
result(" at (x, y)= "+xb+", "+yb)

    Image ADJResult
    ADJResult=ImageCrop(ImageAdj(ResultB, MaxB, RadiusA, CutOff), RadiusA, CutOff)
    SetColorMode(ADJResult, 4)
```

```
      ADJResult.SetName("White BG (%)_Set "+(SetsCount+1))

      If (maxb<=1)
                {
                          SetLimits(ResultB, 0, 1)
                }
      ADJResult.ShowImage()
      ResultB.showimage()
      }

      IntDiffOption2.DLGGetValue(DefaultIntDiff2)
      If(DefaultIntDiff2==1)
      {
      Result("\n\n> Intensity difference A-B")
      Number MaxBB, MinBB, xBB, yBB
      MaxBB=max(Temp_A, xBB, yBB)
      Result("\nMax in template A= "+maxBB)
      result(" at (x, y)= "+xBB+", "+yBB)
      MinBB=min(Temp_A, xBB, yBB)
      Result("\nMin in template A= "+MinBB)
      result(" at (x, y)= "+xBB+", "+yBB)

      MaxBB=max(Temp_B, xBB, yBB)
      Result("\nMax in template B= "+maxBB)
      result(" at (x, y)= "+xBB+", "+yBB)
      MinBB=min(Temp_B, xBB, yBB)
      Result("\nMin in template B= "+MinBB)
      result(" at (x, y)= "+xBB+", "+yBB)

      MaxB=max(ResultBB, xb, yb)
      Result("\nMax= "+maxb)
      result(" at (x, y)= "+xb+", "+yb)
      MinBB=min(ResultBB, xBB, yBB)
      Result("\nMin in correlation map= "+MinBB)
      result(" at (x, y)= "+xBB+", "+yBB)

      Image ADJResultBB
      ADJResultBB=ImageCrop(ImageAdj(ResultBB, MaxB, RadiusA, CutOff), RadiusA, CutOff)
      SetColorMode(ADJResultBB, 4)
      ADJResultBB.SetName("White BG (A-B)_Set "+(SetsCount+1))

      ADJResultBB.ShowImage()
      ResultBB.showimage()
      }

      SetsCount++
      }
      Result("\n\nAverage          "+CrossCorrelationCoeff/Sets+"            "+rFactor/Sets)
      /*
  Result("\n> R-factor: "+rFactor/Sets)
      Result("\n> Cross-correlation coefficient: "+CrossCorrelationCoeff/Sets)
      */

 }

//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//                                                            ////////////////////////////////////////////////////
//   Symmetry quantification for multislices             //////////////////////////////////////////////////////
//                                                            ////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////

 void f_MultiSlices(object self)
 {
      Self.DLGGetValue("CutOffRange", CutOff)
  Image Front:=GetFrontImage() // Base image for calculation
      ImageID=GetImageID(Front)

      Try
      {
                Get3DSize (Front, xpix, ypix, Slices)
      }
      Catch
      {
                OkDialog("This script only works on multi slices.")
                Exit(0)
      }

      DLGGetValue (Value5, xPoints)
      DLGGetValue (Value6, yPoints)

      Result("\n------------------------------------------------------------------------\n")
  Result("\n              CBED symmety check              \n")
  Result("     Kyouhyun Kim, Ph.D candidate, MatSE, U of Illinois ")
  Result("\n                    2012                    \n")
  Result("\n------------------------------------------------------------------------\n")
  Result("\n*Selected options*")
      Result("\n(1) Cut-off range: "+CutOff+" ("+CutOff+" of ROI's area used for caculation")

      Image TempRfactor := RealImage ("", 4, xPoints, yPoints)
      Image ResultrFactor=TempRfactor
      ResultrFactor.SetName("R-factor")
```

```
	Image TempIntMean := RealImage ("", 4, xPoints, yPoints)
Image ResultIntMean=TempIntMean
ResultIntMean.SetName("Intensity difference ratio (%)")
	Image TempCross := RealImage ("Normalized cross-correlation coefficient", 4, xPoints, yPoints)
Image CrossCorrelation=TempCross
CrossCorrelation.SetName("Normalized cross-correlation coefficient")
	Image TempSSD := RealImage ("", 4, xPoints, yPoints)
	Image SSDImg=TempSSD
	SSDImg.SetName("Sum squared difference")

			//Initialize matrix for annotaion ID save function
	AnnotIDSave=AnnotIDSave*0
	AnnotIDSave2=AnnotIDSave2*0

	// Assign annotations' IDs to AnnotA and AnnotB
	Row=0
	Row2=0
	SetsCount=0
	Sets=1
	Number SetsTemp

	Number i=0
	Number ix=0
NoOfAnnot=CountAnnotations(Front)
	While (i<NoOfAnnot)
{
			TempAnnotID=GetNthAnnotationID(Front, i)
			TypeOFAnnot=AnnotationType(Front, TempAnnotID)
			If(TypeOfAnnot==5||TypeOfAnnot==6)
			{
					If(TypeOfAnnot==6)
					{
					AnnotIDSave.SetPixel(0, row, TempAnnotID)
					row++
					}
					If(TypeOfAnnot==5)
					{
					AnnotIDSave2.SetPixel(0, row2, TempAnnotID)
					row2++
					ix++
					}
					i++
					Sets++
			}
			Else
			{
					i++
			}
	}
	Sets=(Sets-1)/2+0.5*ix
	Result("\n>"+Sets+" set(s) selected")
	Result("\n>"+ix+" discs are placed on the mirror plane")

	rFactor=0
	CrossCorrelationCoeff=0

	DLGGetValue(TwoOption, DefaultTwo)
DLGGetValue(ThreeOption, DefaultThree)
DLGGetValue(FourOption, DefaultFour)
DLGGetValue(SixOption, DefaultSix)
DLGGetValue(MirrorOption, DefaultMirror)

	SetsCountB=0
	Number RotAngleTemp
	Number nxx=0, nyy=0

	number Count=0, n=xPoints-1, z=xPoints-1
	Number x_pos=0, y_pos=0
	Image TempWorkImage
	Image TempScrap:=RealImage("Temp Scrap", 4, xPix, yPix)
	Number TempScrapID

	// corecting starts here
	While (Count<Slices)
	{
			TempWorkImage=Front[0, 0, Count, xpix, ypix, Count+1]
			ScrapCopy(TempWorkImage)
			ScrapPaste(TempScrap)
			ScrapMerge(TempScrap)
			TempScrapID=ImageGetID(TempScrap)
			/*
			Result("\nTempScrapID="+TempScrapID)
			Image TestImg:=GetImageFromID(TempScrapID)
			TestImg.ShowImage()
			*/

			While (SetsCount<Sets) //******************************************************************************
			{
					// Allocate annotations' IDs and get annotaions' info
					If (SetsCount<(Sets-ix))
					{
							AnnotA=AnnotIDSave.GetPixel(0, SetsCount*2+1)
							AnnotB=AnnotIDSave.GetPixel(0, SetsCount*2)
```

```
            GetAnnotationRect(Front, AnnotA, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
            GetAnnotationRect(Front, AnnotB, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx)

            //Calculate Radius & Rotation Angle
            ROIInfo(A_AnnTx, A_AnnTy, A_AnnBx, A_CenterX, A_CenterY, RadiusA)
            ROIInfo(B_AnnTx, B_AnnTy, B_AnnBx, B_CenterX, B_CenterY, RadiusB)
            If (RadiusA!=RadiusB)
            {
                B_AnnTx=B_centerX-RadiusA
                B_AnnTy=B_centery-RadiusA
                B_AnnBx=B_centerX+RadiusA
                B_AnnBy=B_centery+RadiusA
            }
            CreateTextAnnotation(Front, A_CenterY, A_CenterX, "A"+(SetsCount+1))
            CreateTextAnnotation(Front, B_CenterY, B_CenterX, "B"+(SetsCount+1))
            RotAngle=slope(A_CenterX, A_CenterY, B_CenterX, B_CenterY)
            /*
            Result ("\n\nRotation angle="+Rad2Ang(RotAngle)+"\n")
            */
        }
        If (SetsCount>=(Sets-ix))
        {
            If (nxx<1) //nxx calculate rotatiion angle of disc lied on the mirror axis once
            {
                AnnotA=AnnotIDSave2.GetPixel(0, SetsCountB+1)
                AnnotB=AnnotIDSave2.GetPixel(0, SetsCountB)
                /*
                Result("\nRectangle!")
                Result("\nAnnotA="+AnnotA+" AnnotB="+AnnotB)
                */
                GetAnnotationRect(Front, AnnotA, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                GetAnnotationRect(Front, AnnotB, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx)
                ROIInfo(A_AnnTx, A_AnnTy, A_AnnBx, A_CenterX, A_CenterY, RadiusA)
                ROIInfo(B_AnnTx, B_AnnTy, B_AnnBx, B_CenterX, B_CenterY, RadiusB)

                If (RadiusA!=RadiusB)
                {
                    B_AnnTx=B_centerX-RadiusA
                    B_AnnTy=B_centery-RadiusA
                    B_AnnBx=B_centerX+RadiusA
                    B_AnnBy=B_centery+RadiusA
                }

                CreateTextAnnotation(Front, A_CenterY, A_CenterX, "m"+(SetsCountB+2))
                CreateTextAnnotation(Front, B_CenterY, B_CenterX, "m"+(SetsCountB+1))
                RotAngleTemp=slope(A_CenterX, A_CenterY, B_CenterX, B_CenterY) //Calculate rotation
angle first
                SetsCountB=SetsCountB+2
            }
        }

        number Count=0, n=xPoints-1, z=xPoints-1
        Number x_pos=0, y_pos=0

        Image Temp_A, Temp_B, TempMask
        Image TempAAA, TempBBB

        // Rotation symmetry
        If (DefaultTwo==1||DefaultThree==1||DefaultFour==1||DefaultSix==1)
        {
            RotAngle=0
            Image TempSymm_A=Rotation(TempScrapID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngle)
            Temp_A=ImageCrop(TempSymm_A, RadiusA, 1)
            Image TempSymm_B=Rotation(TempScrapID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngle)
            If (DefaultTwo==1)
            {
                RotAngle=Deg2Rad(180)
            }
            If (DefaultThree==1)
            {
                RotAngle=Deg2Rad(120)
            }
            If (DefaultFour==1)
            {
                RotAngle=Deg2Rad(90)
            }
            If (DefaultSix==1)
            {
                RotAngle=Deg2Rad(60)
            }
            Temp_B:=ImageCrop(Rotate(TempSymm_B, RotAngle), RadiusA, 1)
            Temp_A.SetName("Disk A_Set "+(SetsCount+1))
            Temp_B.SetName("Disk B_Set "+(SetsCount+1))
        }

        // Mirror symmetry
        If (DefaultMirror==1)
        {
            //Image rotation
            If (SetsCount>=(Sets-ix))
            {
                TempAAA=Rotation(TempScrapID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngleTemp)
                TempBBB=Rotation(TempScrapID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngleTemp)
```

```
            If (nyy==0)
            {
                Temp_A=TempAAA
                Temp_B=TempAAA
                nxx++
                nyy++
            }
            Else If (nyy==1)
            {
                Temp_A=TempBBB
                Temp_B=TempBBB
                nyy=0
                nxx=0
            }
        }

        Else // If there are no discs on mirror direction.
        {
            Temp_A=Rotation(TempScrapID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx, RotAngle)
            Temp_B=Rotation(TempScrapID, B_AnnTy, B_AnnTx, B_AnnBy, B_AnnBx, RotAngle)
        }
    }

    //Create a mask
    Get2Dsize(Temp_A, xWidth, yHeight)
    TempMask=Mask(TempMask, A_AnnBx, A_AnnTx, xWidth, yHeight, CutOff)
    NoOfPixels=sum(TempMask) //Calculate a number of pixes inside the created mask (to be used for quantification)

    // Generate Mask Image ->  generate mirror image -> apply
    //(4) Mask x Image
    Temp_A=Temp_A*TempMask

    Image Temp_AA, Temp_BB, FlipTemp
    If (DefaultMirror==1)
    {
        If (SetsCount>=(Sets-ix))
        {
            Temp_A.SetName("aligned m_Set "+(SetsCount+1))
            FlipVertical(Temp_B)
            Temp_B=Temp_B*TempMask
            Temp_B.SetName("mirror applied_Set "+(SetsCount+1))
        }

        Else
        {
            Temp_A.SetName("aligned A_Set "+(SetsCount+1))
            FlipHorizontal(Temp_B)
            Temp_B=Temp_B*TempMask
            Temp_B.SetName("symm applied B_set "+(SetsCount+1))
            Temp_BB=Temp_B
            FlipHorizontal(Temp_BB)
            Temp_BB.SetName("aligned B_set "+(SetsCount+1))
        }
    }

    Else
    {
        Temp_B=Temp_B*TempMask
        Temp_B.SetName("Disk B_Set "+(SetsCount+1))
    }

    // r-factor
    Number Temp_rFactor=f_Rfactor(Temp_A, Temp_B, TempMask, xWidth, yHeight)
    rFactor=rFactor+Temp_rFactor


    // Cross-correlation coefficient
    Number Temp_XCross=XCorr(Temp_A, Temp_B, TempMask, NoOfPixels)
    CrossCorrelationCoeff=CrossCorrelationCoeff+Temp_XCross

    /*
    // (7) Contrast difference in %
    Image ResultB=IntRatio(Temp_A, Temp_B, xWidth, yHeight)
    Statistic(ResultB, TempMask, NoOfPixels, mean_A, STDV_A)
    TempIntMean.SetPixel(x_Pos, y_Pos, mean_A)

    // (9) SSD
    SumSquaredDifference=SSD(Temp_A, Temp_B)
    TempSSD.SetPixel(x_Pos, y_Pos, SumSquaredDifference)
    */

    SetsCount++
} // 'while' for SetsCount

SetsCount=0 // Initialize SetsCount to calculate the selected discs on the next slide
SetsCountB=0 // SetsCountB is used for the annotaion ID of rectangulars (initialize)

TempRfactor.SetPixel(x_Pos, y_Pos, rFactor)
TempCross.SetPixel(x_Pos, y_Pos, CrossCorrelationCoeff)
rFactor=0 // Initialize rFactor for the next slide
CrossCorrelationCoeff=0 // Initialize xCross for the next slide

Count++
```

```
                OpenAndSetProgressWindow("Calculating slice "+(Count)+"...",round(Count/Slices*100)+"% completed","")
                x_Pos=x_Pos+1

                //Move the position in the result image
                if (Count>z)
                {
                          y_Pos=y_Pos+1
                          z=z+xPoints
                          x_Pos=0
                }

      } // End of Loop while

      Number MaxVal, MinVal, xMax, yMax, xMin, yMin

      Result("\n1. R-factor map\n")
      ResultrFactor=TemprFactor/Sets
      ResultrFactor.ShowImage()
      /*
      ResultrFactor.SetColorMode(4)
      */
      MaxMin(ResultrFactor, MaxVal, xMax, yMax, MinVal, xMin, yMin)
      Result("Best= "+MinVal+" at ("+xMin+", "+yMin+")\n")
      Result("Worst= "+MaxVal+" at ("+xMax+", "+yMax+")\n")

      Result("\n2. Normalized cross-correlation coefficient map\n")
      CrossCorrelation=TempCross/Sets
      CrossCorrelation.ShowImage()
      /*
      CrossCorrelation.SetColorMode(4)
      */
      MaxMin(CrossCorrelation, MaxVal, xMax, yMax, MinVal, xMin, yMin)
      Result("Best= "+MaxVal+" at ("+xMax+", "+yMax+")\n")
      Result("Worst= "+MinVal+" at ("+xMin+", "+yMin+")\n")

      /*
      Result("\n3. Intensity diference ratio (%) map\n")
      ResultIntMean=ResultIntMean/Sets
      ResultIntMean.ShowImage()
      ResultIntMean.SetColorMode(4)
      MaxMin(ResultIntMean, MaxVal, xMax, yMax, MinVal, xMin, yMin)
      Result("Best= "+MinVal+" at ("+xMin+", "+yMin+")\n")
      Result("Worst= "+MaxVal+" at ("+xMax+", "+yMax+")\n")

      Result("\n4. Sum squared difference map\n")
      SSDImg=SSDImg/Sets
      SSDImg.ShowImage()
      SSDImg.SetColorMode(4)
      MaxMin(SSDImg, MaxVal, xMax, yMax, MinVal, xMin, yMin)
      Result("Best= "+MinVal+" at ("+xMin+", "+yMin+")\n")
      Result("Worst= "+MaxVal+" at ("+xMax+", "+yMax+")\n")
      */

}

// Image extraction******************************************************************************
 void f_Extract(object self)
 {
      DLGGetValue (Value5, xPoints)
  DLGGetValue(Value7, xCor)
  DLGGetValue(Value8, yCor)
  Image TempA=GetImageFromID(ImageID)
      GetSize(TempA, xPix, yPix)

      Number SliceNo
      SliceNo=xCor+xPoints*yCor

      Image TempB=TempA[0, 0, SliceNo, xPix, yPix, SliceNo+1]
      TempB.SetName("Slice "+SliceNo)
  TempB.ShowImage()
 }


// Save templates******************************************************************************
 void f_SaveCoord(object self)
 {
      DLGGetValue (Coord1, DefaultCoord1)
      DLGGetValue (Coord2, DefaultCoord2)
      DLGGetValue (Coord3, DefaultCoord3)

      Image Front:=GetFrontImage()

      Row=0
      Number i=0
  NoOfAnnot=CountAnnotations(Front)

  If (DefaultCoord1==1)
  {
                MatCoord1=0
                While (i<NoOfAnnot)
                {
                          TempAnnotID=GetNthAnnotationID(Front, i)
                          TypeOFAnnot=AnnotationType(Front, TempAnnotID)
```

```
            If(TypeOfAnnot==5||TypeOfAnnot==6)
            {
                If(TypeOfAnnot==5)
                {
                    MatCoord1.SetPixel(4, row, 1)
                }
                Else If(TypeOfAnnot==6)
                {
                    MatCoord1.SetPixel(4, row, 0)
                }
                GetAnnotationRect(Front, TempAnnotID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                MatCoord1.SetPixel(0, row, A_AnnTy)
                MatCoord1.SetPixel(1, row, A_AnnTx)
                MatCoord1.SetPixel(2, row, A_AnnBy)
                MatCoord1.SetPixel(3, row, A_AnnBx)
                row++
            }
            i++
        }
    row=0
    }

 If (DefaultCoord2==1)
 {
        MatCoord2=0
        While (i<NoOfAnnot)
        {
            TempAnnotID=GetNthAnnotationID(Front, i)
            TypeOFAnnot=AnnotationType(Front, TempAnnotID)
            If(TypeOfAnnot==5||TypeOfAnnot==6)
            {
                If(TypeOfAnnot==5)
                {
                    MatCoord2.SetPixel(4, row, 1)
                }
                Else If(TypeOfAnnot==6)
                {
                    MatCoord2.SetPixel(4, row, 0)
                }
                GetAnnotationRect(Front, TempAnnotID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                MatCoord2.SetPixel(0, row, A_AnnTy)
                MatCoord2.SetPixel(1, row, A_AnnTx)
                MatCoord2.SetPixel(2, row, A_AnnBy)
                MatCoord2.SetPixel(3, row, A_AnnBx)
                row++
            }
            i++
        }
    row=0
    }

    If (DefaultCoord3==1)
 {
        MatCoord3=0
        While (i<NoOfAnnot)
        {
            TempAnnotID=GetNthAnnotationID(Front, i)
            TypeOFAnnot=AnnotationType(Front, TempAnnotID)
            If(TypeOfAnnot==5||TypeOfAnnot==6)
            {
                If(TypeOfAnnot==5)
                {
                    MatCoord3.SetPixel(4, row, 1)
                }
                Else If(TypeOfAnnot==6)
                {
                    MatCoord3.SetPixel(4, row, 0)
                }
                GetAnnotationRect(Front, TempAnnotID, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                MatCoord3.SetPixel(0, row, A_AnnTy)
                MatCoord3.SetPixel(1, row, A_AnnTx)
                MatCoord3.SetPixel(2, row, A_AnnBy)
                MatCoord3.SetPixel(3, row, A_AnnBx)
                row++
            }
            i++
        }
    row=0
    }

    }//Close save coordinates function


// Load templates********************************************************************
void f_LoadCoord(object self)
{
    DLGGetValue (Coord1B, DefaultCoord1B)
    DLGGetValue (Coord2B, DefaultCoord2B)
    DLGGetValue (Coord3B, DefaultCoord3B)

    Image Front:=GetFrontImage()

    Number i=0
```

```
   If (DefaultCoord1B==1)
   {
                         While (1)
                         {
                                   A_AnnTy=GetPixel(MatCoord1, 0, i)
                                   A_AnnTx=GetPixel(MatCoord1, 1, i)
                                   A_AnnBy=GetPixel(MatCoord1, 2, i)
                                   A_AnnBx=GetPixel(MatCoord1, 3, i)
                                   If(A_AnnTy==0&&A_AnnTx==0&&A_AnnBy==0&&A_AnnBx==0)
                                   {
                                             exit(0)
                                   }
                                   If (GetPixel(MatCoord1, 4, i)==0)
                                   {
                                             CreateOvalAnnotation(Front, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                                   }
                                   Else If (GetPixel(MatCoord1, 4, i)==1)
                                   {
                                             CreateBoxAnnotation(Front, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                                   }
                                   i++
                         }
       i=0
       }

   If (DefaultCoord2B==1)
   {
                         While (1)
                         {
                                   A_AnnTy=GetPixel(MatCoord2, 0, i)
                                   A_AnnTx=GetPixel(MatCoord2, 1, i)
                                   A_AnnBy=GetPixel(MatCoord2, 2, i)
                                   A_AnnBx=GetPixel(MatCoord2, 3, i)
                                   If(A_AnnTy==0&&A_AnnTx==0&&A_AnnBy==0&&A_AnnBx==0)
                                   {
                                             exit(0)
                                   }
                                   If (GetPixel(MatCoord2, 4, i)==0)
                                   {
                                             CreateOvalAnnotation(Front, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                                   }
                                   Else If (GetPixel(MatCoord2, 4, i)==1)
                                   {
                                             CreateBoxAnnotation(Front, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                                   }
                                   i++
                         }
       i=0
       }

   If (DefaultCoord3B==1)
   {
                         While (1)
                         {
                                   A_AnnTy=GetPixel(MatCoord3, 0, i)
                                   A_AnnTx=GetPixel(MatCoord3, 1, i)
                                   A_AnnBy=GetPixel(MatCoord3, 2, i)
                                   A_AnnBx=GetPixel(MatCoord3, 3, i)
                                   If(A_AnnTy==0&&A_AnnTx==0&&A_AnnBy==0&&A_AnnBx==0)
                                   {
                                             exit(0)
                                   }
                                   If (GetPixel(MatCoord3, 4, i)==0)
                                   {
                                             CreateOvalAnnotation(Front, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                                   }
                                   Else If (GetPixel(MatCoord3, 4, i)==1)
                                   {
                                             CreateBoxAnnotation(Front, A_AnnTy, A_AnnTx, A_AnnBy, A_AnnBx)
                                   }
                                   i++
                         }
       i=0
       }
       }//Close load coordinates function
}//Close class uiframe

//End of function
//------------------------------------------------------------------------
//------------------------------------------------------------------------



// Creates the buttons, fields and labels in the dialog

//Script Information
taggroup ScriptInfo()
{
       taggroup ScriptInfo=dlgcreatelabel("\nSymmetry quantification of CBED pattern")
       ScriptInfo.dlgexternalpadding(50,5)
       return ScriptInfo
}
```

```
TagGroup OptionBox()
{
	taggroup box1_items
	taggroup box1=dlgcreatebox("  Options  ", box1_items)

	TagGroup CutName1=DLGCreateLabel(" Cut-off range:    0<")
	CutName1.DLGExternalPadding(0, 4)
	TagGroup CutOffRange=DLGCreateRealField(DefaultCutOff,7,4) // DLGCreateRealField(Number Value, Number Width, Number Format)
	CutOffRange.DlgExternalPadding(0,0)
	CutOffRange.DLGIdentifier("CutOffRange")
	TagGroup CutName2=DLGCreateLabel("<1                         ")
	CutName2.DLGExternalPadding(1,1)
	TagGroup CutGroup=DLGGroupItems(CutName1, CutOffRange, CutName2)
	CutGroup.DLGTableLAyout(3, 1, 0)
	CutGroup.DLGExternalPadding(0, 0)
	Box1_Items.DLGAddElement(CutGroup)

	LegendOption=DLGCreateCheckBox("Legend (Yellow=100%, Black=0%)", DefaultLegend)
	LegendOption.DLGExternalPadding(0, 10, 0, 40)
	Box1_items.dlgaddelement(LegendOption)

	TagGroup SymmetryName=DLGCreateLabel("\nSymmetry quantification options:")
	SymmetryName.DLGExternalPadding(0, 0, 2, 83)
	Box1_Items.DLGADDElement(SymmetryName)

	TwoOption=DLGCreateCheckBox("2-fold", DefaultTwo)
	TwoOption.DLGExternalPadding(0, 0, 0, 20)
	ThreeOption=DLGCreateCheckBox("3-fold", DefaultThree)
	ThreeOption.DLGExternalPadding(0, 0, 0, 20)
	FourOption=DLGCreateCheckBox("4-fold", DefaultFour)
	FourOption.DLGExternalPadding(0, 0, 0, 20)
	TagGroup Symmetry_1=DLGGroupItems(TwoOption, ThreeOption, FourOption)
	Symmetry_1.DLGTableLayout(3, 1, 0)
	Symmetry_1.DLGExternalPadding(0,0, 0, 20)
	Box1_Items.DLGADDElement(Symmetry_1)

	SixOption=DLGCreateCheckBox("6-fold", DefaultSix)
	SixOption.DLGExternalPadding(0, 0, 0, 20)
	MirrorOption=DLGCreateCheckBox("mirror", DefaultMirror)
	MirrorOption.DLGExternalPadding(0, 0, 0, 20)
	TagGroup Symmetry_2=DLGGroupItems(SixOption, MirrorOption)
	Symmetry_2.DLGTableLayout(2, 1, 0)
	Symmetry_2.DLGExternalPadding(0,0, 0, 90)
	Box1_Items.DLGADDElement(Symmetry_2)

	return box1
}


TagGroup MainControlBox1()
{
	// Make Buttons
	taggroup box1_items
	taggroup box1=dlgcreatebox("  Alignment  ", box1_items)

	//f_AutoFind
	TagGroup MenuName3=DLGCreateLabel("1. Alignment option")
	MenuName3.DLGExternalPadding(0, 0, 2, 143)
	Box1_Items.DLGADDElement(MenuName3)
	/*
	AdjIntRatio=DLGCreateCheckBox("Min intensity difference ratio", DefaultIntRatio)
	AdjIntRatio.DLGExternalPadding(0, 0, 2, 80)
	Box1_items.dlgaddelement(AdjIntRatio)
	*/
	AdjRfactor=DLGCreateCheckBox("Min R-factor", DefaultRfactor)
	AdjRfactor.DLGExternalPadding(0, 0, 2, 153)
	Box1_items.dlgaddelement(AdjRfactor)
	AdjXcorr=DLGCreateCheckBox("Max normalized cross-correlation coeff", DefaultXCorr)
	AdjXcorr.DLGExternalPadding(0, 0, 2, 30)
	Box1_items.dlgaddelement(AdjXcorr)


	TagGroup Label1 = DLGCreateLabel("x: ")
	Label1.DlgExternalPadding(1,1)
	Value1=DLGCreateIntegerField(0, 7)
	Value1.DlgExternalPadding(0,0)
	DlgValue(Value1, xMove)
	TagGroup Label2 = DLGCreateLabel("points with step size (pix) ")
	Label2.DlgExternalPadding(1,1)
	Value2=DLGCreateIntegerField(0, 7)
	Value2.DlgExternalPadding(0,0)
	DlgValue(Value2, xStep)
	TagGroup Group2=DLGGroupItems(Label1, Value1, Label2, value2)
	Group2.DLGTableLayout(4, 1, 0)
	Group2.DLGExternalPadding(0,0)
	Box1_Items.DLGADDElement(Group2)

	TagGroup Label3 = DLGCreateLabel("y: ")
	Label3.DlgExternalPadding(1,1)
	Value3=DLGCreateIntegerField(0, 7)
	Value3.DlgExternalPadding(0,0)
	DlgValue(Value3, yMove)
```

```
    TagGroup Label4 = DLGCreateLabel("points with step size (pix) ")
    Label4.DlgExternalPadding(1,1)
    Value4=DLGCreateIntegerField(0, 7)
    Value4.DlgExternalPadding(0,0)
    DlgValue(Value4, yStep)
    TagGroup Group3=DLGGroupItems(Label3, value3, label4, value4)
    Group3.DLGTableLayout(4, 1, 0)
    Group3.DLGExternalPadding(0,0)
    Box1_Items.DLGADDElement(Group3)

    TagGroup AutoFindButton=DLGCreatePushButton("   Search   ","f_AutoFind").DLGSide("Center")
    AutoFindButton.dlgexternalpadding(0,0)
    Box1_Items.DLGADDElement(AutoFindButton)

    TagGroup Label5=DLGCreateLabel("2. Save templates")
    Label5.DlgExternalPadding(0,0,2,140)
    Box1_items.dlgaddelement(Label5)
    Coord1=DLGCreateCheckBox("1", DefaultCoord1)
    Coord1.DLGExternalPadding(0, 0, 2, 20)
    Coord2=DLGCreateCheckBox("2", DefaultCoord2)
    Coord2.DLGExternalPadding(0, 0, 2, 20)
    Coord3=DLGCreateCheckBox("3", DefaultCoord3)
    Coord3.DLGExternalPadding(0, 0, 2, 20)
    TagGroup SaveCoordButton=DLGCreatePushButton("     Save     ", "f_SaveCoord").DLGSide("Center")
    SaveCoordButton.dlgexternalpadding(0,0)
    TagGroup Group5=DLGGroupItems(Coord1, Coord2, Coord3, SaveCoordButton)
    Group5.DLGTableLAyout(4, 1, 0)
    Group5.DLGExternalPadding(0,0)
    Box1_items.dlgaddelement(Group5)

    return box1
}

TagGroup MainControlBox2()
{



    // Control box 2 items
    taggroup box2_items
    taggroup box2=dlgcreatebox("  Symmetry quantification  ", box2_items)

    //Load templates
    TagGroup Label9=DLGCreateLabel("1. Load templates")
    Label9.DlgExternalPadding(0,0,2,150)
    Box2_items.dlgaddelement(Label9)
    Coord1B=DLGCreateCheckBox("1", DefaultCoord1B)
    Coord1B.DLGExternalPadding(0, 0, 2, 20)
    Coord2B=DLGCreateCheckBox("2", DefaultCoord2B)
    Coord2B.DLGExternalPadding(0, 0, 2, 20)
    Coord3B=DLGCreateCheckBox("3", DefaultCoord3B)
    Coord3B.DLGExternalPadding(0, 0, 2, 20)
    TagGroup LoadCoordButton=DLGCreatePushButton("     Load     ", "f_LoadCoord").DLGSide("Center")
    LoadCoordButton.dlgexternalpadding(0,0)
    TagGroup Group6=DLGGroupItems(Coord1B, Coord2B, Coord3B, LoadCoordButton)
    Group6.DLGTableLAyout(4, 1, 0)
    Group6.DLGExternalPadding(0,0)
    Box2_items.dlgaddelement(Group6)

    //f_SingleImage
    TagGroup MenuName1=DLGCreateLabel("\n2. Quantification for a disp image  ")
    MenuName1.dlgexternalpadding(1,1)
    TagGroup SingleImageButton = DLGCreatePushButton("   Run   ", "f_SingleImage").DLGSide("Center")
    SingleImageButton.dlgexternalpadding(0,0,-11, 11)
    TagGroup group1 = DLGGroupItems(MenuName1, SingleImageButton)
    Group1.DLGTableLayout(2, 1, 0)
    box2_items.dlgaddelement(Group1)
    group1.dlgexternalpadding(0,0)
    TemplateOption=DLGCreateCheckBox("Show calculated templates", DefaultTemplate)
    TemplateOption.DLGExternalPadding(0, 0, 2, 85)
    Box2_items.dlgaddelement(TemplateOption)
    IntDiffOption1=DLGCreateCheckBox("Intensity map (%)", DefaultIntDiff1)
    IntDiffOption1.DLGExternalPadding(0, 0, 2, 2)
    IntDiffOption2=DLGCreateCheckBox("Intensity map (counts)", DefaultIntDiff2)
    IntDiffOption2.DLGExternalPadding(0, 0, 2, 3)
    TagGroup IntGroup=DLGGroupItems(IntDiffOption1, IntDiffOption2)
    IntGroup.DLGTableLayout(2, 1, 0)
    IntGroup.DLGExternalPadding(0,0)
    Box2_items.dlgaddelement(IntGroup)


    // f_MultiSlices
    TagGroup MenuName4=DLGCreateLabel("\n3. Quantification for multislices                          ")
    MenuName4.dlgexternalpadding(1,1)
    Box2_items.DLGAddElement(MenuName4)
    TagGroup label5 = DLGCreateLabel("x points:")
    label5.dlgexternalpadding(1,1)
    Value5 = DLGCreateIntegerField(0, 5)
    Value5.DlgExternalPadding(0,0)
    DlgValue(Value5,xPoints)

    TagGroup label6 = DLGCreateLabel(" y points:")
    label6.dlgexternalpadding(1,1)
```

```
        Value6 = DLGCreateIntegerField(0, 5)
        Value6.DLGexternalpadding(0,0)
        DlgValue(Value6,yPoints)
        TagGroup SlicesGroup1=DLGGroupItems(label5, Value5, label6, Value6)
        SlicesGroup1.DLGTableLayout(4, 1, 0)

        TagGroup SlicesButton = DLGCreatePushButton("   Run   ", "f_MultiSlices").DLGSide("Center")
        SlicesButton.dlgexternalpadding(0,-5, 0, 0)

        TagGroup SlicesGroup2=DLGGroupItems(SlicesGroup1, SlicesButton)
        SlicesGroup2.DLGTableLayout(2, 1, 0)
        Box2_Items.DLGADDElement(SlicesGroup2)
        SlicesGroup2.dlgexternalpadding(0,3)

        TagGroup Label7=DLGCreateLabel("Extract image at (x, y)")
        Label7.DLGExternalPadding(1,1)
        Value7=DLGCreateIntegerField(0, 5)
        DLGValue(Value7, xCor)
        Value7.DLGexternalpadding(0,0)
        Value8=DLGCreateIntegerField(0, 5)
        DLGValue(Value8, yCor)
        Value8.DLGexternalpadding(0,0)
        TagGroup ExtractButton = DLGCreatePushButton("Extract", "f_Extract").DLGSide("Center")
        ExtractButton.dlgexternalpadding(0, -2, 0, 0)
        TagGroup ExtractGroup=DLGGroupItems(Label7, Value7, Value8, ExtractButton)
        ExtractGroup.DLGTableLayout(4, 1, 0)
        Box2_Items.DLGADDElement(ExtractGroup)

        return box2
}


//My information
taggroup MyInfo()
        {
                        taggroup MyInfo=dlgcreatelabel("Kyouhyun Kim, MatSE, U of Illinois, 2012\n")
                        MyInfo.dlgexternalpadding(0,0)
                        return MyInfo
        }
//End


//
// Function to assemble the compenents of the dialog into the dialog frame

void DialogLibraryIntensitySum()
{
        // Configure the positioning in the top right of the application window

        TagGroup position;
        position = DLGBuildPositionFromApplication()
        position.TagGroupSetTagAsTagGroup( "Width", DLGBuildAutoSize() )
        position.TagGroupSetTagAsTagGroup( "Height", DLGBuildAutoSize() )
        position.TagGroupSetTagAsTagGroup( "X", DLGBuildRelativePosition( "Inside", 1) )
        position.TagGroupSetTagAsTagGroup( "Y", DLGBuildRelativePosition( "Inside", -1.0 ) )

        TagGroup dialog_items;
        TagGroup dialog = DLGCreateDialog("SED", dialog_items).dlgposition(position);


        // Call each button
        dialog_items.DLGAddElement(ScriptInfo())
        dialog_items.DLGAddElement(OptionBox())
   dialog_items.DLGAddElement(MainControlBox1())
   dialog_items.DLGAddElement(MainControlBox2())
   dialog_items.DLGAddElement(MyInfo())


        object dialog_frame = alloc(DialogLibraryTestClass4).init(dialog)
        dialog_frame.display("Symmetry quantification");
}


// Call the main function which creates and displays the dialog

DialogLibraryIntensitySum()
```