

State-of-the-art Texture Analysis Plugin of SmartLab Studio II

Akihiro Himeda, Akito Sasaki, Keigo Nagao, Mike Rusetsky

Rigaku Corporation

himeda@rigaku.co.jp

In this work we will present the *Texture* plugin of *SmartLab Studio II* which is intended for quantitative texture analysis. This plugin is a complete solution that includes both input data processing and orientation distribution function (ODF) reconstruction.

As the input data for texture analysis software can use pole figures which are measured by the conventional technique using a 0D detector or by the more advanced technique using a 2D detector. This software uses a powerful module for data from 2D detectors which uses integrated diffraction pattern fitting technique for calculation of diffraction peak intensity. That allows accurate background estimation and ability to separate closely spaced peaks. Several types of corrections are also available for processing of input pole figures.

This plugin implements two methods of ODF reconstruction: WIMV and components methods. Both methods could be used for all types of crystal lattice symmetry and 2 types of sample symmetry – triclinic and orthorhombic. Also, this plugin can use the most popular definitions, in texture community, of Eulerian angles: Bunge notation, Roe/Matthiers notation.

WIMV method is a direct (pure numerical) method of ODF reconstruction. It calculates values of function on discrete grid in 3-dimensional rotation space. From a user point of view, this is the simplest method because it needs only a few parameters for calculation. Realization of this method also contains iteration of ODF background. Component method is more complicated to use, but gives more representative result. *Texture* plugin implements 3 types of components: spherical (peak), axial (fiber) and elliptical. Software contains database of components and user can manage this database to add or remove components. Realization of method includes two minimization algorithms – Genetic algorithm and Nelder-Mead. Due to stochastic nature of Genetic algorithm it allows (in many cases) to find valid component position even for badly defined initial approximation.

