

# GrainSpotter Version 0.3 - User manual

Søren Schmidt

[soeren.schmidt@risoe.dk](mailto:soeren.schmidt@risoe.dk)

October 15, 2007

## **Abstract**

This is a preliminary version of the documentation for GrainSpotter and is by no means complete. Nevertheless, the current version should provide the user with enough information to run GrainSpotter.

# Chapter 1

## Short description of the algorithm

GrainSpotter is a pattern recognition algorithm for identifying crystallographic orientations (grains) in polycrystalline materials. Based on simulations hundreds of grains can be recovered simultaneously. GS takes as input a list of measurements (g-vectors in reciprocal space) and a list of theoretical scatter vectors derived from the lattice of the material in question. The algorithm then finds replica of the set of theoretical scatter vectors in the measured data. The discrete set of rotations that brings the set of theoretical scatter vectors onto a subset of the measured g-vectors are the crystallographic orientations of the grains present in the material.

### 1.1 Algorithm

Any rotation can be expressed as a rotation axis,  $\bar{n}$  (3 by 1 vector), and a rotation around the rotation axis,  $w$  (scalar) (e.g. in the Rodrigues formulation one has  $\bar{r} = \bar{n} \tan(\frac{w}{2})$ ). Consequently, for all vectors  $\bar{v} = \bar{g} - \bar{t}$  belonging to the orientation, where  $\bar{g}$  is the measured scatter vector and  $\bar{t}$  is corresponding theoretical scatter vector, we have

$$\bar{n} \cdot \bar{v} \approx 0. \quad (1.1)$$

If two pairs  $(\bar{g}_1, \bar{t}_1)$  and  $(\bar{g}_2, \bar{t}_2)$  of measured and theoretical scatter vectors have the same internal angle,  $\bar{g}_1 \cdot \bar{g}_2 = \bar{t}_1 \cdot \bar{t}_2$ , the rotation axis is given by

$$\bar{n} = \bar{v}_1 \times \bar{v}_2. \quad (1.2)$$

The properties (1.1) and (1.2) are used throughout the GS algorithm. For each sub region of the orientation space GS finds candidate rotation axes fulfilling (1.2). For each rotation axis additional measurements that fulfill (1.1) are collected. A new rotation axis is then fitted based on these measurements (2 by 2 matrices). Afterwards, the rotation angles,  $w$ 's, are calculated and a median filter applied. This gives a candidate orientation  $\bar{r} = \bar{n} \tan(w/2)$  and a search for additional measurements are done by forward calculations of the remaining theoretical scatter vectors. The final orientation is given

by a fit to all the selected measurements, fitting both  $n$  and  $w$  simultaneously (3 by 3 matrices). A few properties of the GS implementation

- GS makes use of error propagation. If individual measurement errors are provided for each g-vector, these errors are carried through the algorithm thus favoring accurate measurements. E.g. (1.1) reads  $|\bar{n} \cdot \bar{v}| < N_{\sigma} \sigma_n$ , where  $\sigma_n$  is the error on the direction of the rotations axis.
- Pseudo twin identification. In, especially, highly symmetrical space groups there are often more than one crystallographic solution to a set of reciprocal vectors (pseudo-twin ambiguity problem). GS always finds the maximal solution among the pseudo twins.
- All rotations of theoretical vectors and search for measurements are done using integer calculations.

## Chapter 2

# Using GrainSpotter

GS operates with three text files

- initialization file (input file, e.g. data.ini)
- data file, also called g-vector file (input file, e.g. data.gve)
- grains file (output file, e.g. data.glog)

The standalone version of GS is launched from the prompt:

```
GrainSpotter.0.3 data.ini
```

GS reads the initialization and g-vector files and processes the data. Afterwards, the recovered crystallographic orientations are written to the grains file.

### 2.1 Initialization file

The initialization file consists of a set of *cards*. Each card is followed by a list of arguments:

**card name (argument1/units) ... (argumentN/units)**

Currently, the initialization file holds 10 cards:

**spacegroup (number/integer)**

**filespecs (g-vector input filename) (grains output filename)**

**etarange (begin/degrees) (end/degrees)** Can appear multiple times, one for each  $\eta$ -range.

**domega (delta/degrees)** The integration range in  $\omega$  by which the diffraction images were recorded.

**omegarange (begin/degrees) (end/degrees)** Can appear multiple times, one for each  $\omega$ -range.

**cuts (min. number of measurements/integer) (min. completeness/float)**

**eulerstep (step size in orientation space/degrees)**

**uncertainties ( $\sigma_{2\theta}$ /degrees) ( $\sigma_\eta$ /degrees) ( $\sigma_\omega$ /degrees)**

**nsigmas ( $N_\sigma$ /float)**

**N\_hkls\_in\_indexing (Number of rings used in indexing/integer)**

## 2.2 g-vector file

The g-vector file is produced by ImageD11, see related specifications on the file format (<http://fable.sourceforge.net/index.php/ImageD11>).

## 2.3 grains file

The format of the grains file is a list of recovered crystallographic orientations. Each grain is listed with the following information:

$G_{id}$ : Grain id

$N_{exp}$ -**gvectors**  $N_{meas}$ -**gvectors**: Number of expected and measured scatter vectors.

$\bar{\mu}_{IA}$ : Average internal angle in degrees between the predicted and measured scatter vector.

$U_{11}U_{12}U_{13}$

$U_{21}U_{22}U_{23}$

$U_{31}U_{32}U_{33}$ : Orientation maxtrix

$g_{id} \ hkl \ 2\theta_{meas} \ 2\theta_{pred} \ \Delta 2\theta \ \omega_{meas} \ \omega_{pred} \ \Delta\omega \ \eta_{meas} \ \eta_{pred} \ \Delta\eta \ IA$