

A”

Aalto University
School of Engineering

COE-C2004 - Materials Science and Engineering

Exercise 1

Prof. Junhe Lian

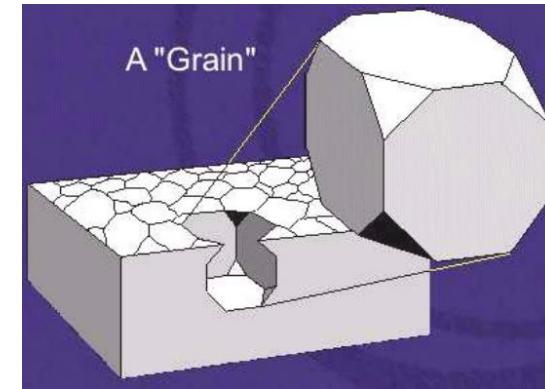
Wenqi Liu (Teaching assistant)

Outline

- Introduction
 - Microstructure
 - Investigation techniques
 - Crystallographic orientation
 - Grain size and shape
- Software introduction
- Assignments and questions

Microstructure characterization

Crystalline Materials - Microstructure



[http://engineering.snu.ac.kr/lecture/texture&anisotropy/Texture%20&%20Anisotropy%2010\(Representation\).pdf](http://engineering.snu.ac.kr/lecture/texture&anisotropy/Texture%20&%20Anisotropy%2010(Representation).pdf)

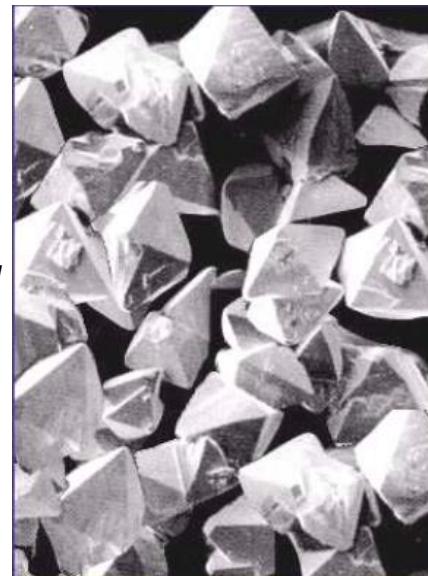
Microstructure

The arrangement of the constituents of a material, i.e. the spatial distribution of elements, phases, orientation, and defects. [1]

Length scale: nm-cm for microstructure
pm for crystal structure

Phase in material science

A phase is a region of material that shows the uniform chemical and structure.



Crystal/Grain

A grain is a region of discrete crystal orientation within a polycrystalline material.

- Most crystalline solids are composed of a collection of many small crystals or grains; such materials are termed as *polycrystalline*.
- In metals and other materials, the individual grains may fit together closely to form the solid.

Investigation techniques - Light Optical Microscopy

- Uses light – usually up to 2000X magnification with resolution $\geq 0.3 \mu\text{m}$
- Grinding and polishing to remove surface features (e.g., scratches)
- Etching by special solutions to change reflectance, depending on grain orientation, chemical composition, etc.
- Ambiguous identification of phases
- No precise crystallographic information

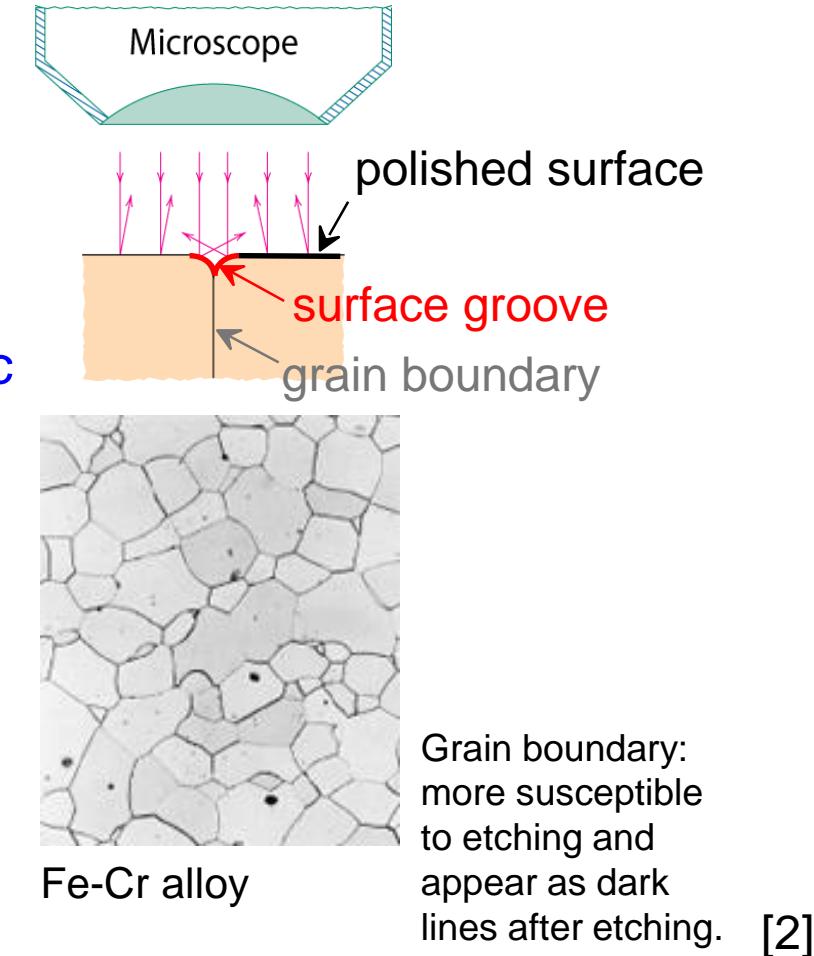
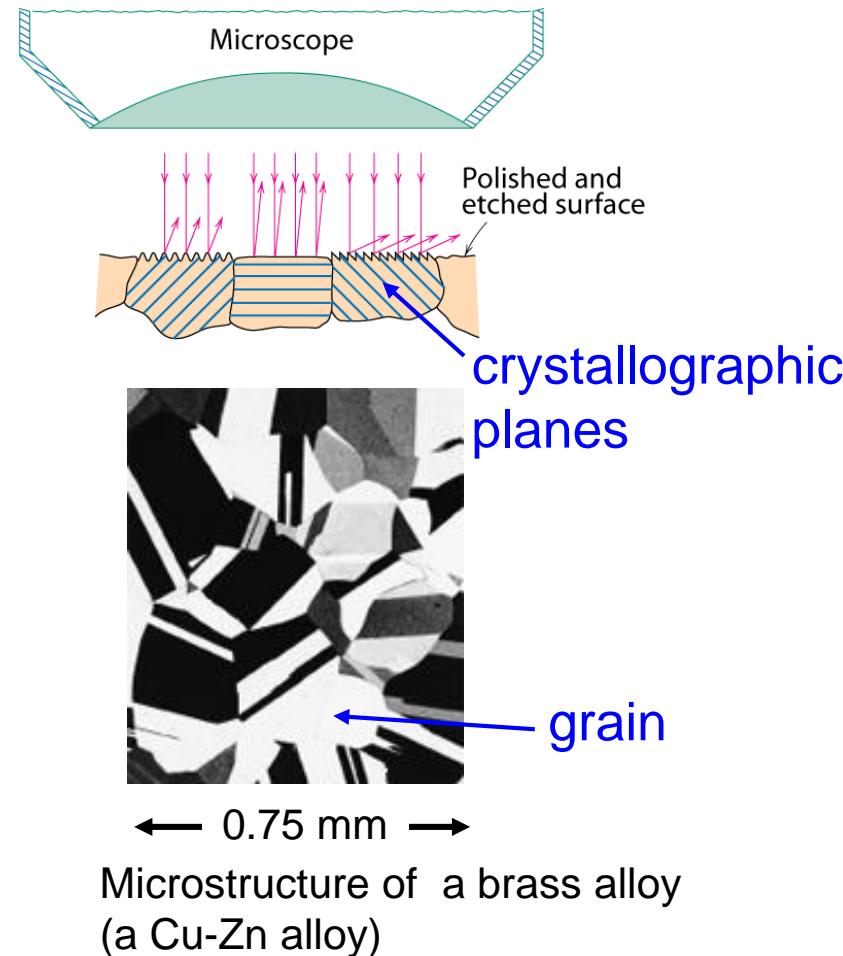
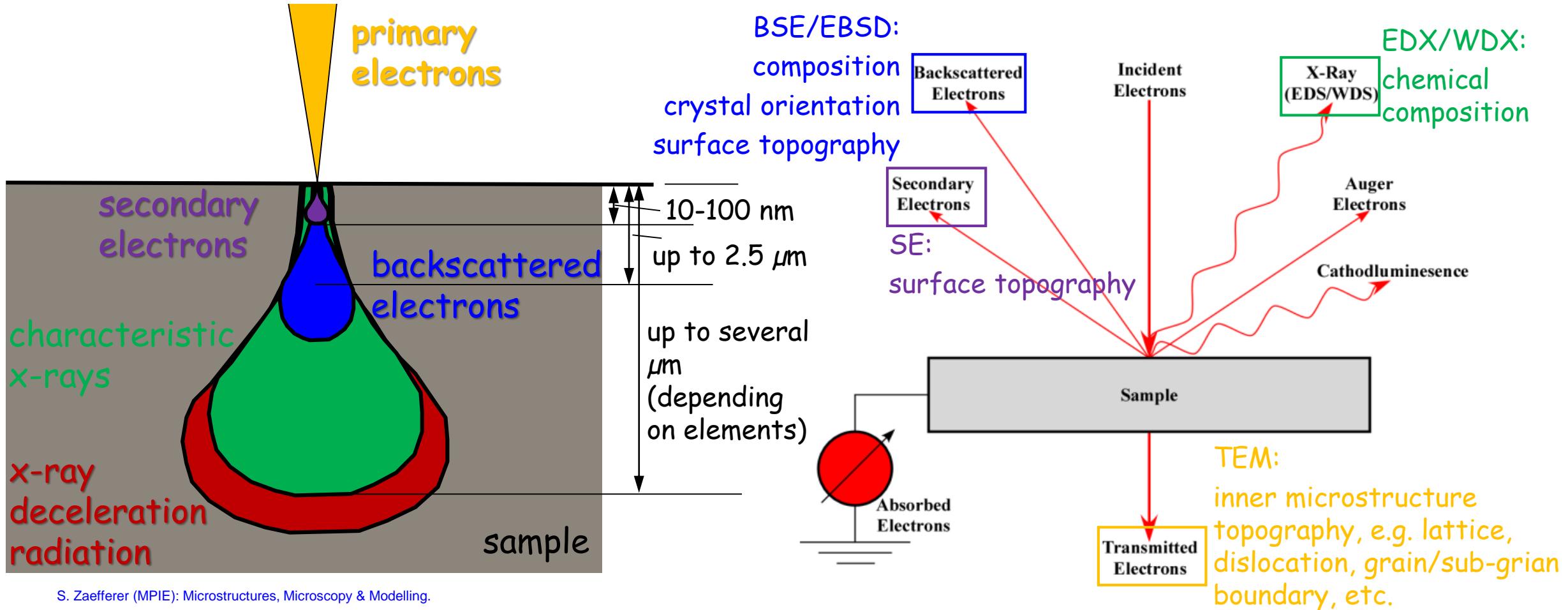


Fig. 4.14(b) & (c), Callister & Rethwisch 10e.

Fig. 4.15(a) & (b), Callister & Rethwisch 10e.

Investigation techniques - Electrical Microscopy

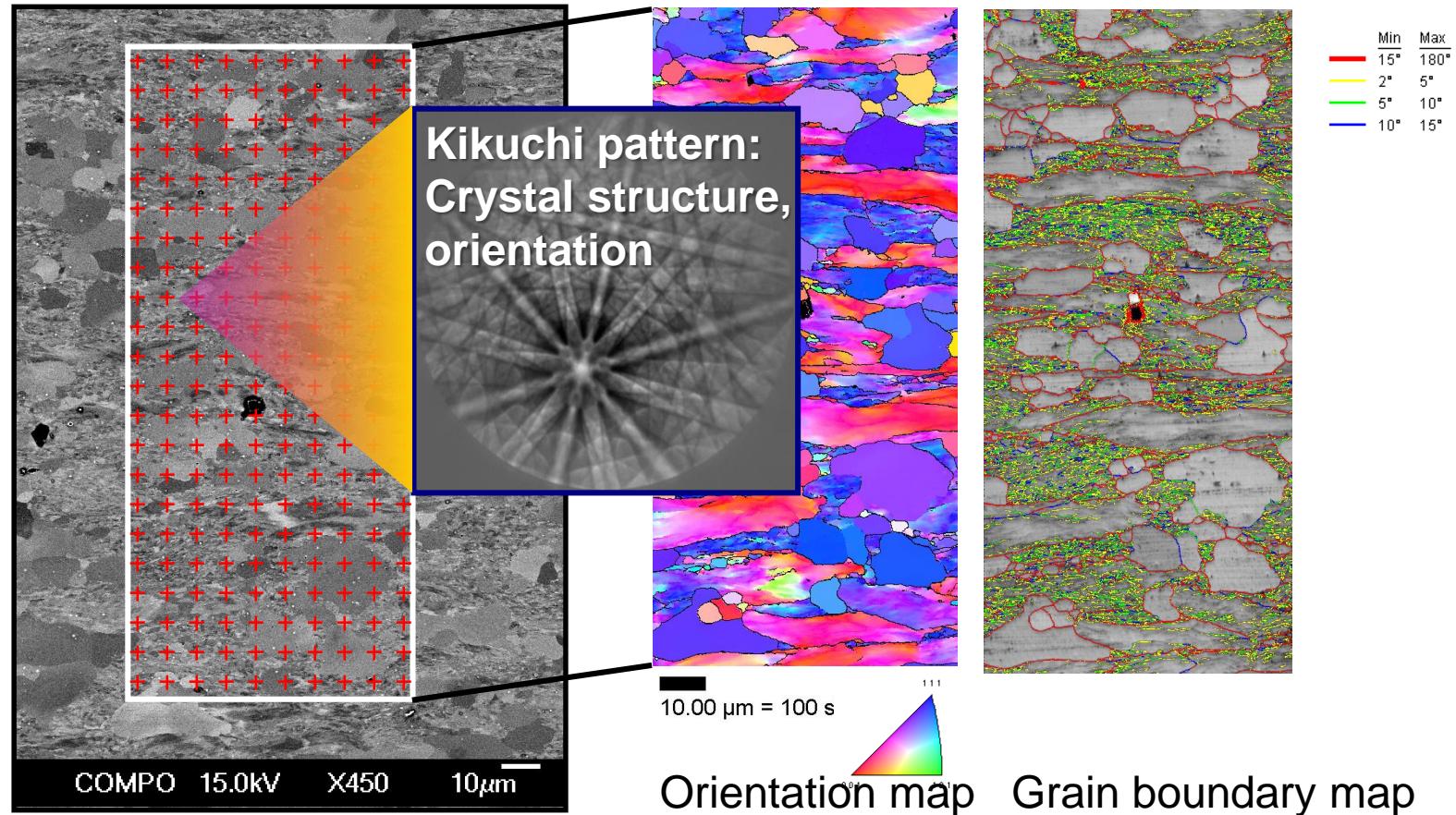
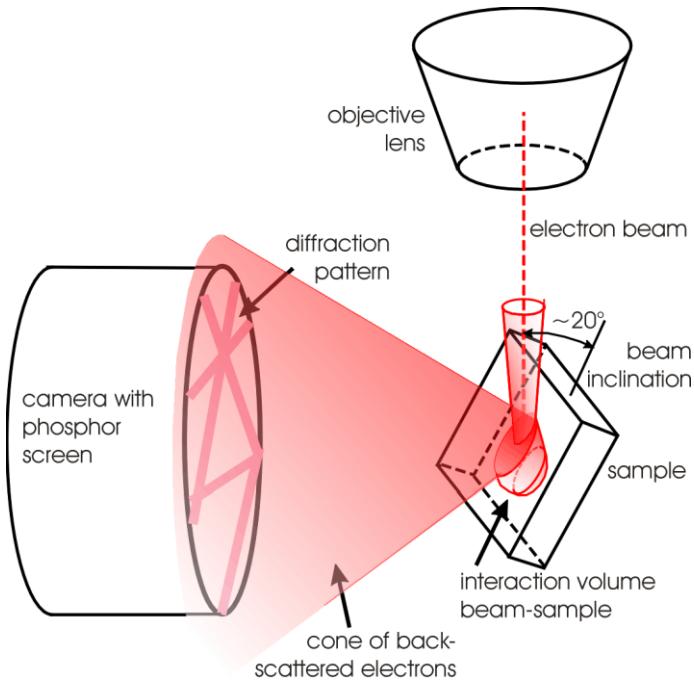
Reaction and interaction of electron beam encountering a solid -> Different signals -> Different detectors



S. Zaefferer (MPIE): Microstructures, Microscopy & Modelling.

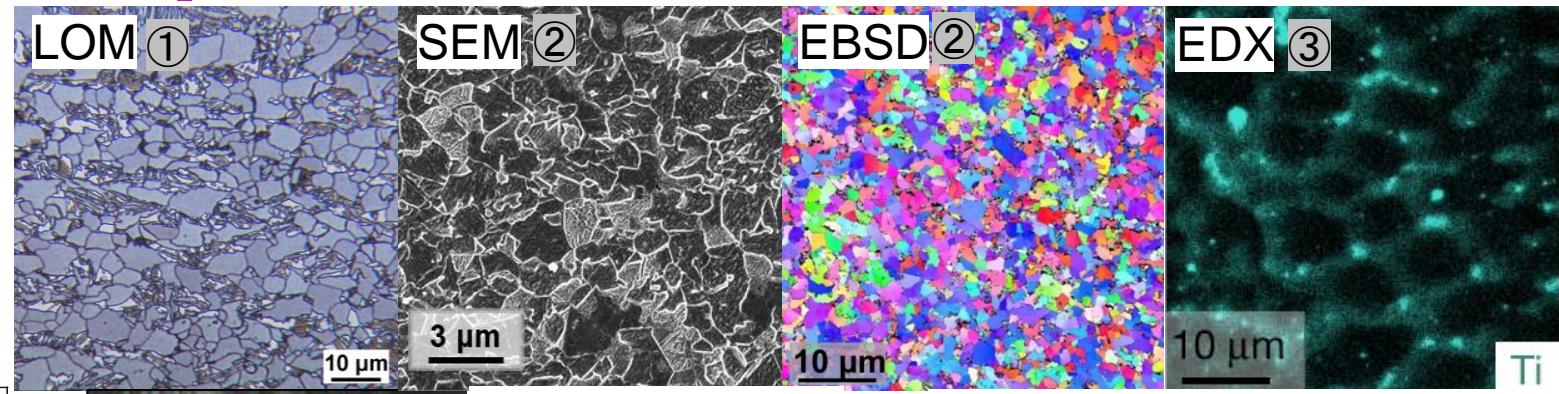
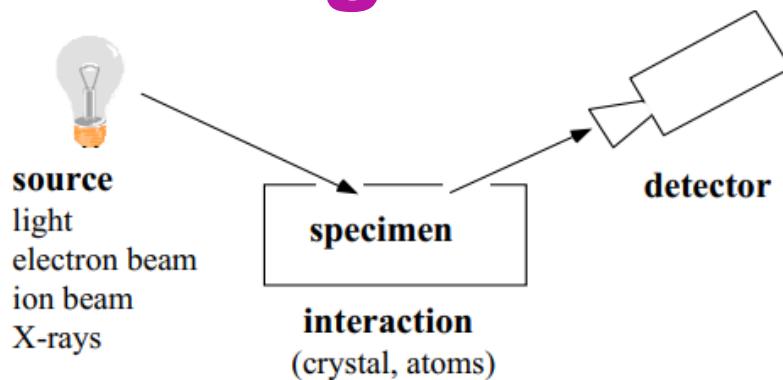
Investigation techniques - EBSD

EBSD - Electron Backscatter Diffraction

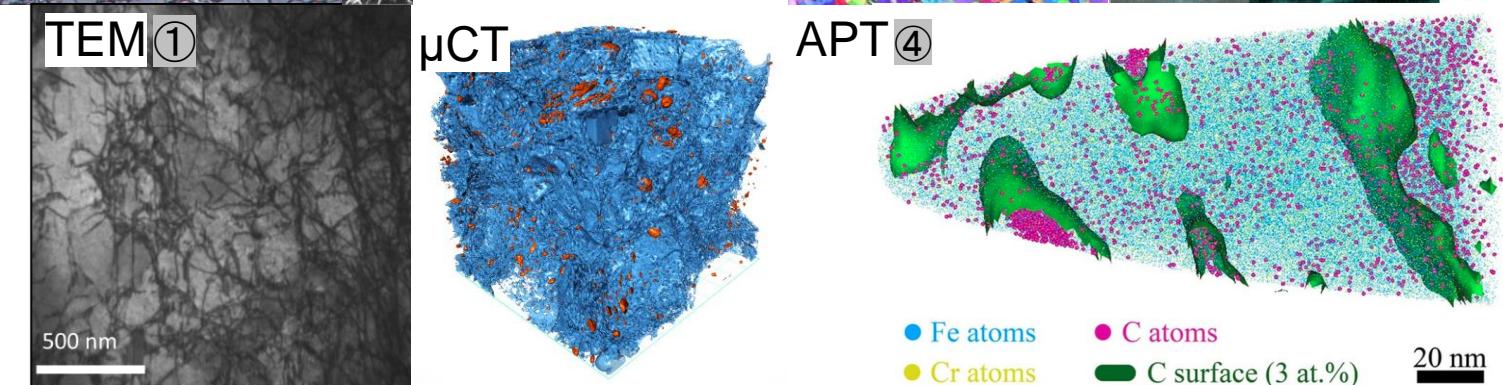


S. Zaefferer (MPIE): Microstructures, Microscopy & Modelling.

Investigation techniques



Method	LOM	FIB	SEM	TEM	XRD
source	Light	Ion	Electron	Electron	X-rays
image resolution	300nm	20nm	2nm	0.1nm	(mm)
imaging	yes	yes	yes	yes	no
crystal structure	no	no	EBSD	SAD	yes
chemical composition	no	SIMS	EDX, WDX	EDX, EELS	not directly

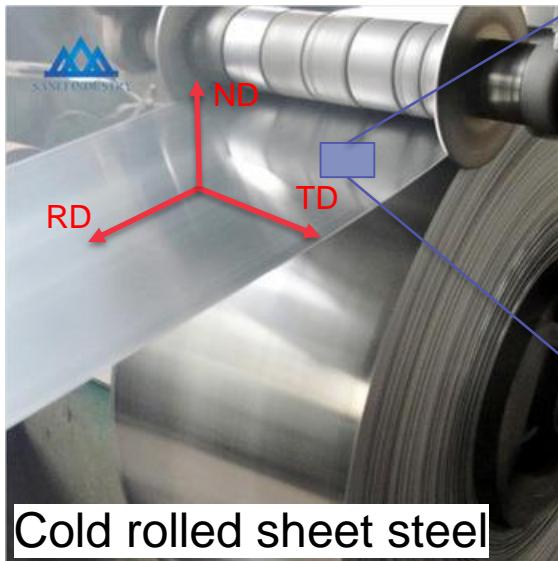


LOM - Light Optical Microscopy; **FIB** - Focused Ion Beam; **SEM** - Scanning Electron Microscopy; **TEM** - Transmission Electron Microscopy; **XRD** - X-ray Diffraction; **EBS** - Electron Backscatter Diffraction; **SAD** - Selected Area Diffraction; **SIMS** - Secondary Ion Mass Spectrometry; **EDX** - Energy Dispersive X-ray Spectroscopy; **WDX** - Wavelength Dispersive X-ray Spectroscopy; **EELS** - Electron Energy Loss Spectroscopy; **μCT** - micro Computed Tomography; **APT** - atom probe tomography

C. Scheu (MPIE): Electron Microscopy & Analytical Techniques.

- ① S. Zaeferer (MPIE): Microstructures, Microscopy & Modelling.
 ② <https://www.sciencedirect.com/science/article/pii/S0749641919303729>
 ③ <https://doi.org/10.1038/s41586-020-2409-3>
 ④ <http://www.dierk-raabe.com/atom-probe-tomography/>

Crystallographic orientation

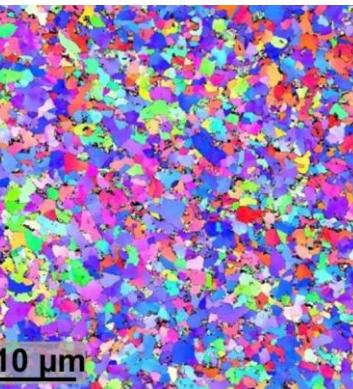


<https://images.app.goo.gl/eSLBdrEBpMiqqqpr5>

RD: Rolling Direction

TD: Transverse Direction

ND: Normal Direction

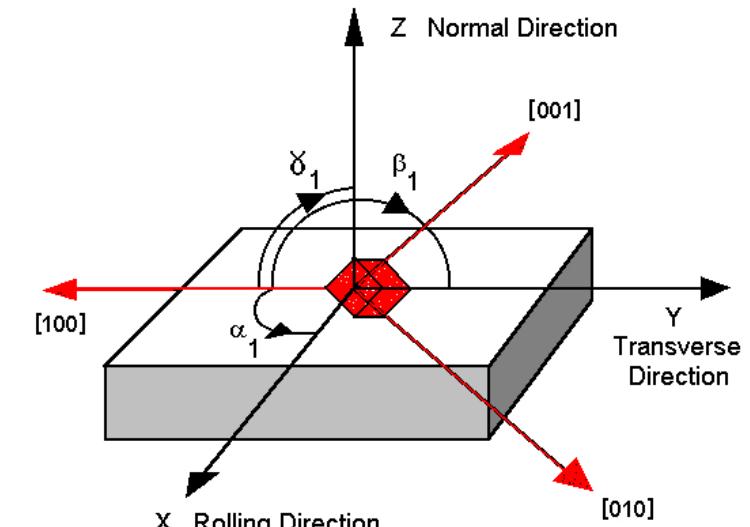
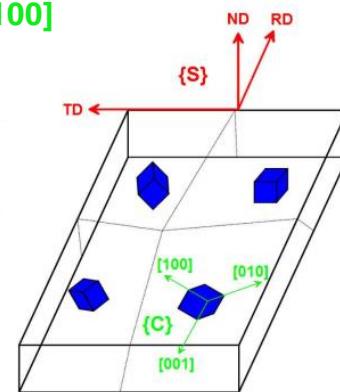
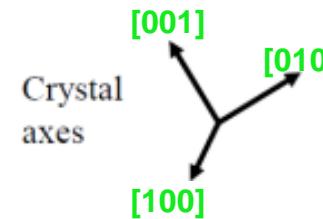
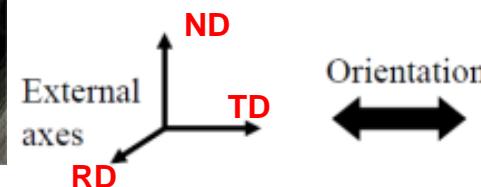


Crystallographic orientation:

Rotation g to transform the sample frame $\{S\}$ into the crystal frame $\{C\}$: $\{C\} = g \cdot \{S\}$

g : 3x3 rotation matrix

mathematical description $g = \begin{pmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{pmatrix} = \begin{pmatrix} \cos\alpha_1 & \cos\beta_1 & \cos\gamma_1 \\ \cos\alpha_2 & \cos\beta_2 & \cos\gamma_2 \\ \cos\alpha_3 & \cos\beta_3 & \cos\gamma_3 \end{pmatrix}$



Olaf Engler (Hydro Aluminium): Introduction to Texture Analysis: Macrotexture, Microtexture, Orientation Microscopy

[3]

Crystallographic orientation

Miller-indices method (hkl) $[uvw]$

$(hkl)[uvw]$: $(hkl) \parallel ND$, $[uvw] \parallel RD$

either normalized to unity or, preferably, smallest whole number

$$g = \begin{pmatrix} u/N_1 & q/N_2 & h/N_3 \\ v/N_1 & r/N_2 & k/N_3 \\ w/N_1 & s/N_2 & l/N_3 \end{pmatrix} \quad N \text{ for normalization}$$

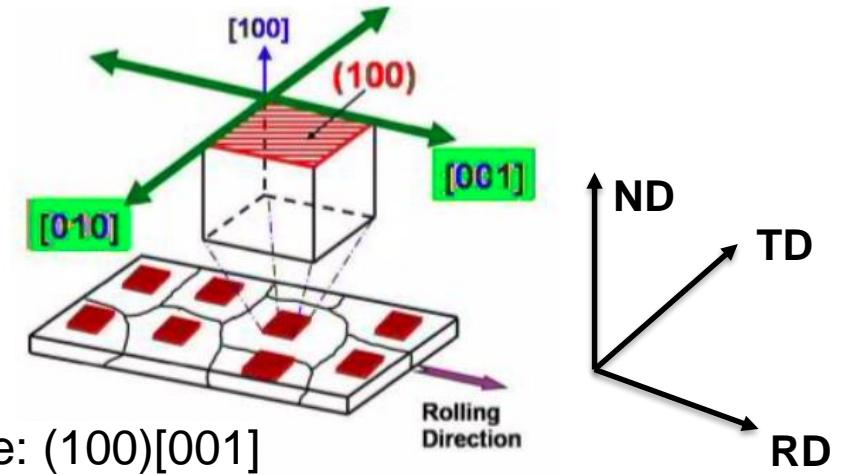
(hkl) vector of ND in $\{C\}$

$[qrs]$ vector of TD in $\{C\}$

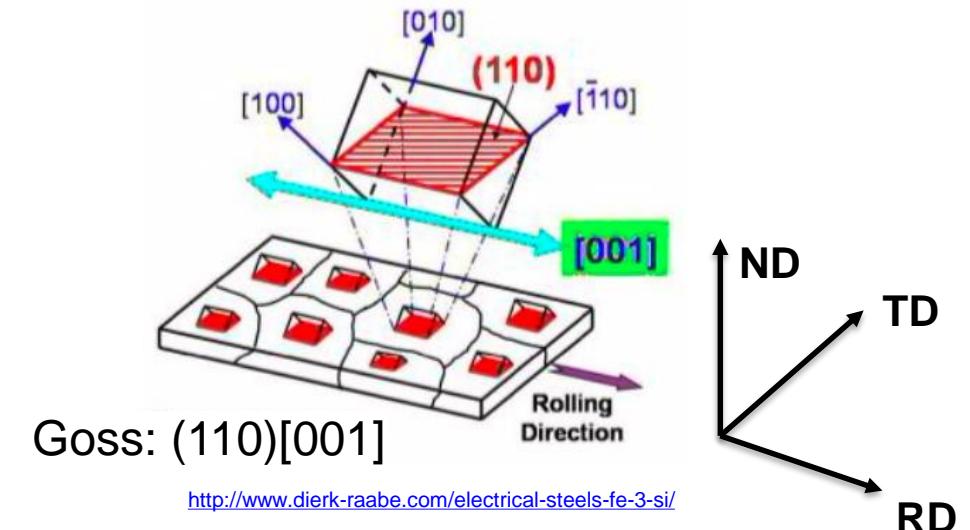
$[uvw]$ vector of RD in $\{C\}$

$$\begin{pmatrix} q \\ r \\ s \end{pmatrix} = \begin{pmatrix} h \\ k \\ l \end{pmatrix} \times \begin{pmatrix} u \\ v \\ w \end{pmatrix}$$

[3]



Cube: $(100)[001]$



Goss: $(110)[001]$

<http://www.dierk-raabe.com/electrical-steels-fe-3-si/>

Crystallographic orientation

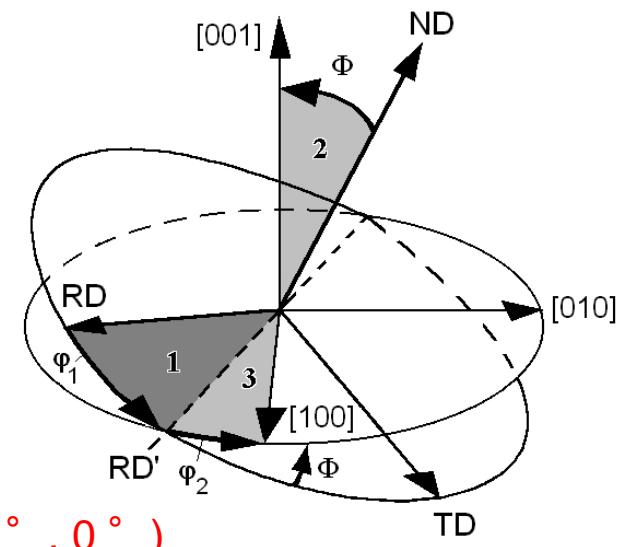
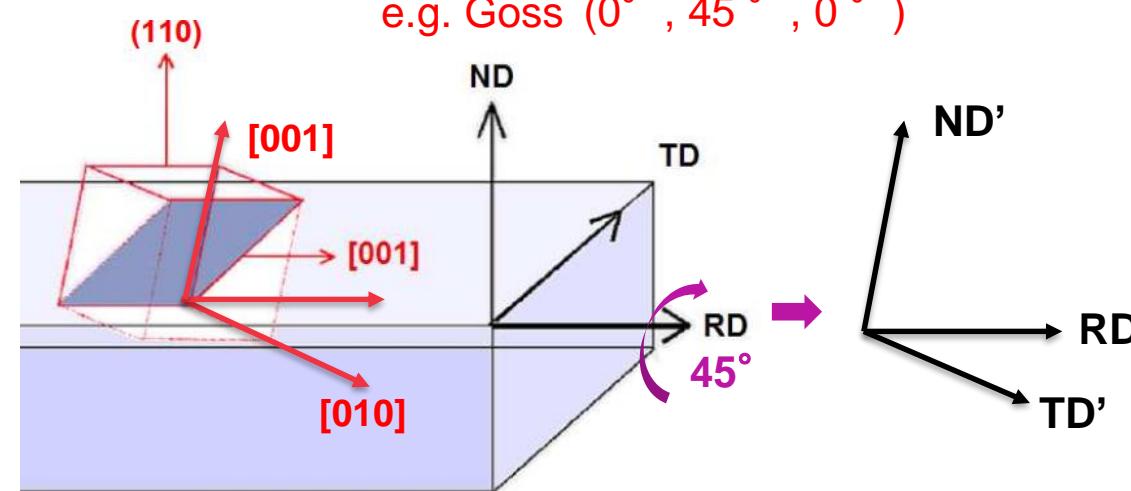
Euler angle method (φ_1 , Φ , φ_2)

Bunge method (Bunge, 1969): Rotate the sample frame {S} into the crystal frame {C}.

1. rotation by φ_1 about the normal direction ND, transforming the rolling direction RD into RD', on (001) plane of the crystal frame;
2. rotation by Φ about the axis RD' (in its new orientation), transforming the normal direction ND into direction [001] of the crystal frame and bringing TD into the (001) plane in the crystal frame;
3. rotation by φ_2 about ND" (in its new orientation, i.e. [001] direction).

$$g = g_{\varphi_2} \cdot g_{\Phi} \cdot g_{\varphi_1}$$

$$\left\{ \begin{array}{l} g_{\varphi_1} = \begin{pmatrix} \cos\varphi_1 & \sin\varphi_1 & 0 \\ -\sin\varphi_1 & \cos\varphi_1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ g_{\Phi} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & -\sin\Phi & \cos\Phi \end{pmatrix} \\ g_{\varphi_2} = \begin{pmatrix} \cos\varphi_2 & \sin\varphi_2 & 0 \\ -\sin\varphi_2 & \cos\varphi_2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{array} \right.$$



Crystallographic orientation

Rotation axis/angle method $[uvw]/\theta$

Rotate by angle θ about common axis $[uvw]$

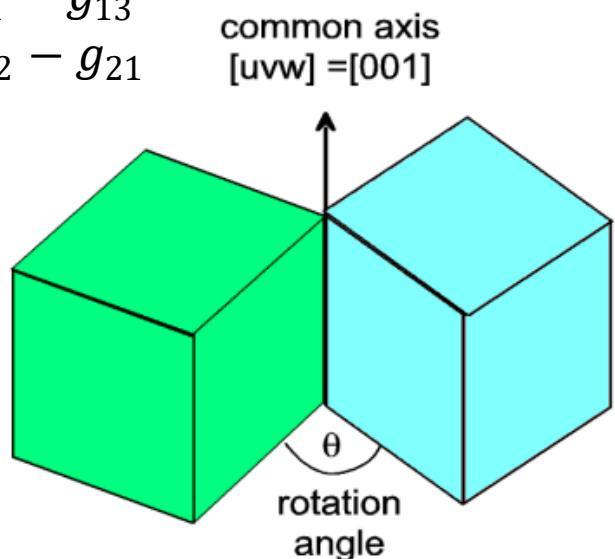
The axis is identical in both crystals

$$2\cos\theta = g_{11} + g_{22} + g_{33} - 1$$

$$u \cdot 2\sin\theta = g_{23} - g_{32}$$

$$v \cdot 2\sin\theta = g_{31} - g_{13}$$

$$w \cdot 2\sin\theta = g_{12} - g_{21}$$



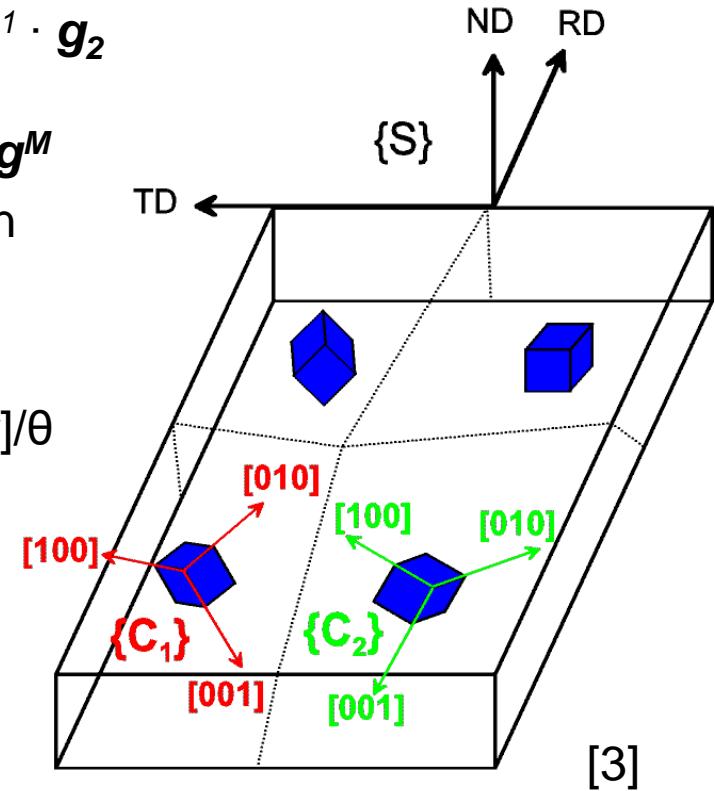
→ Misorientation between two grains

Rotation matrix g^M to transform crystal frame $\{C_1\}$ of grain 1 into the crystal frame $\{C_2\}$ of grain 2:

$$\{C_2\} = g^M \cdot \{C_1\} \text{ or } g^M = g_1^{-1} \cdot g_2$$

The misorientation matrix g^M is analogous to the rotation matrix g , and it can be expressed in the rotation axis/angle parameter $[uvw]/\theta$ method.

θ : misorientation angle



Grain Boundaries

- regions between grains (crystals)
- crystallographic misalignment across a grain boundary
- Slight atomic disorder
- high atomic mobility
- high chemical reactivity

low-angle grain boundaries (LAGBs) are those with a misorientation less than about 15 degrees. In contrast, the misorientation of *high-angle grain boundaries (HAGBs)* is greater than about 15 degrees.

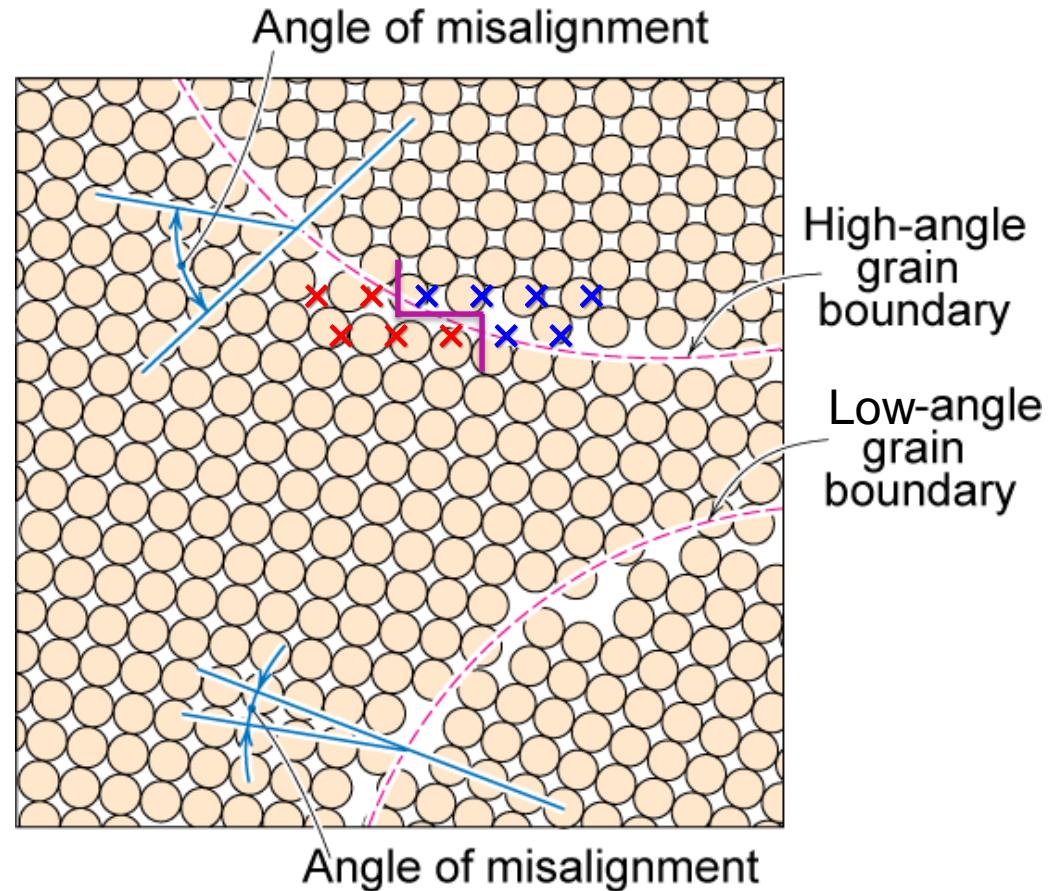


Fig. 4.8, Callister & Rethwisch 10e.

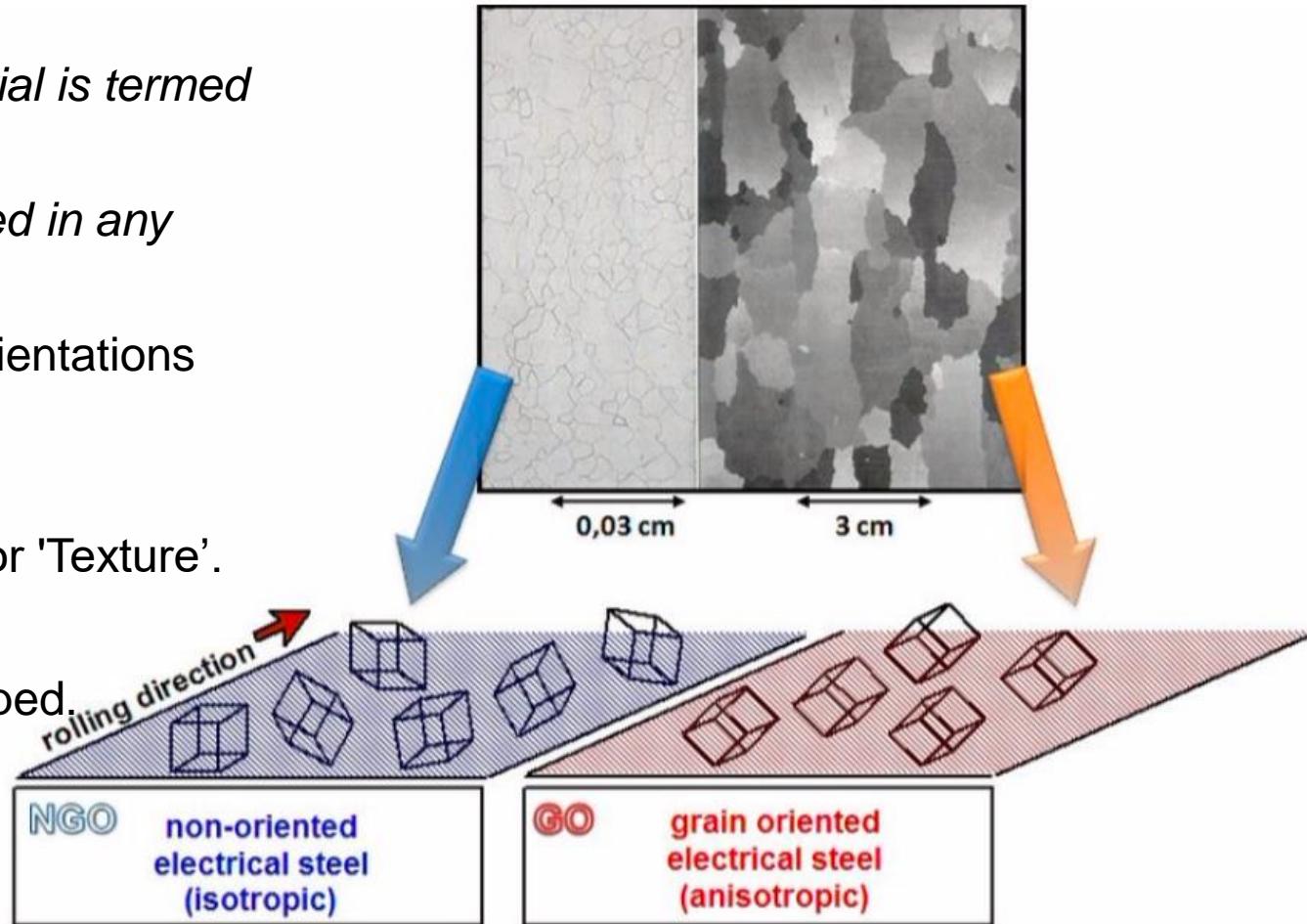
Crystallographic orientation

Isotropic & Anisotropic

- If properties are equal in all directions, a material is termed as 'Isotropic'.
- If the properties tend to be greater or diminished in any direction, a material is termed as 'Anisotropic'.
- In an isotropic polycrystalline material, grain orientations are random.
- Many/most materials are anisotropic.
- Anisotropy results from preferred orientations or 'Texture'.

Texture: preferred orientations in a polycrystal.

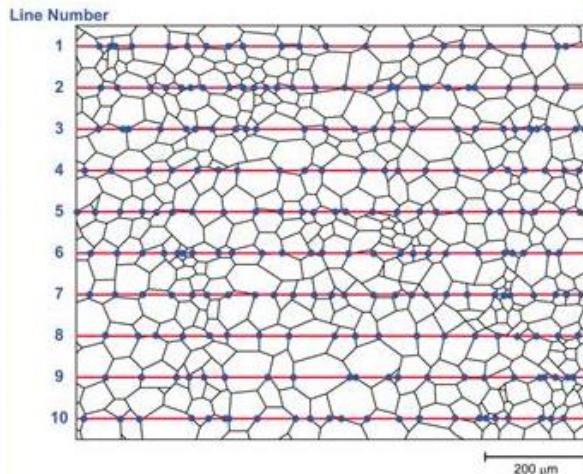
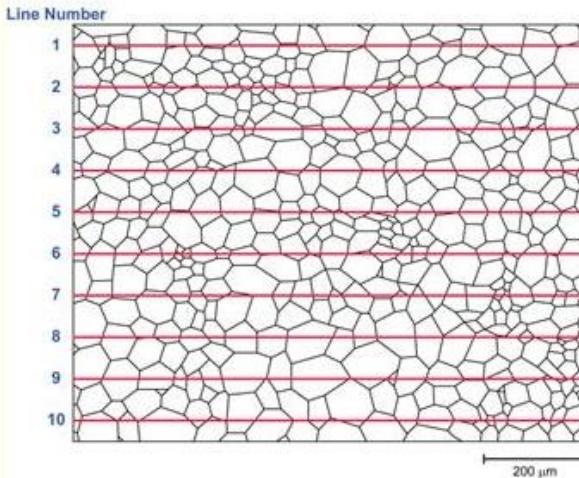
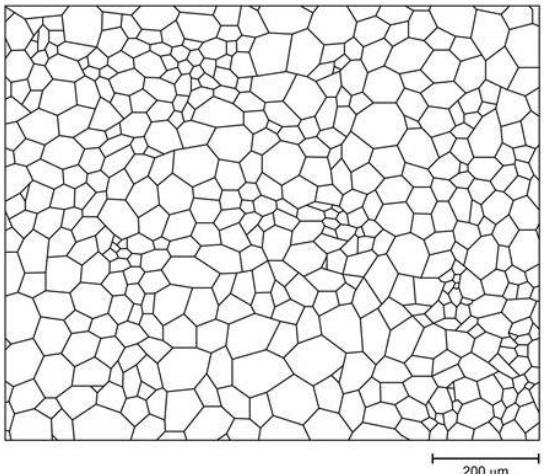
- Texture may range from slight to highly developed.



<http://www.dierk-raabe.com/electrical-steels-fe-3-si/>

Grain size

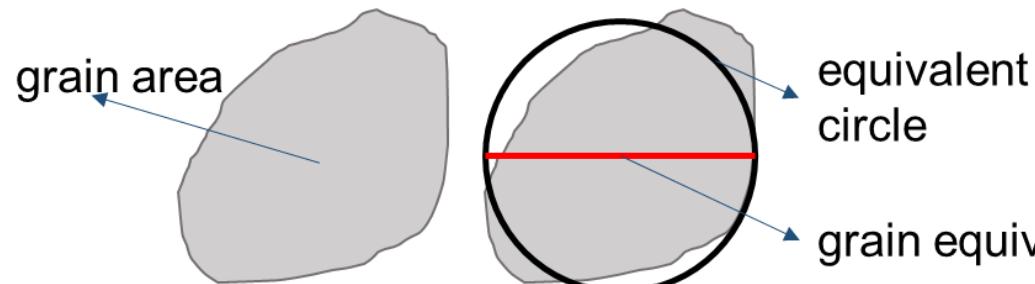
<https://mybiblioteka.su/10-20243.html>



Average grain size

$$= \frac{\text{Total number of intercepts}}{\text{Total line length}}$$

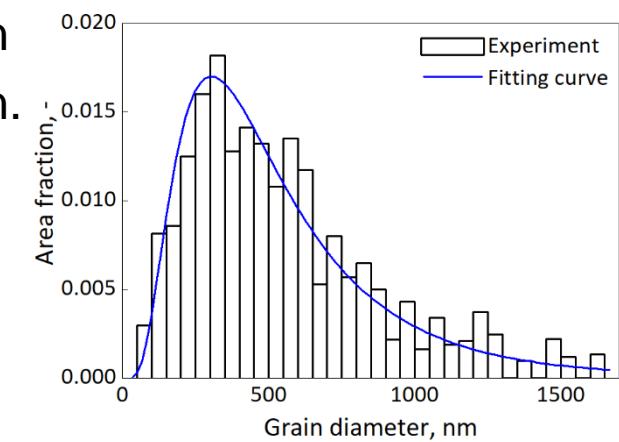
Equivalent grain diameter d



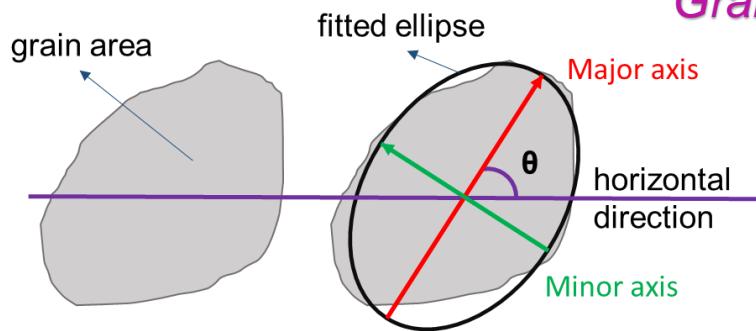
Fit a circle to the points
making up a grain based on
the equivalent area criterion.

$$f(d; \mu, \sigma) dy = \frac{1}{d\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln d - \mu)^2}{2\sigma^2}\right) dy, \quad \text{mean}(d) = \exp(\mu + \sigma^2/2)$$

log-normal distribution



Grain shape



Fit an ellipse to the points making up a grain based on the equivalent area criterion.

Grain shape factor: Equivalent grain aspect ratio

$$\text{Aspect ratio} = \frac{\text{Length of the minor axis}}{\text{Length of the major axis}}$$

Examples:

1:1:1 refers to equiaxed grains,
1:0.1 to cold-rolled grains,
ranges from 0 to 1.

Grain shape angle/ orientation θ

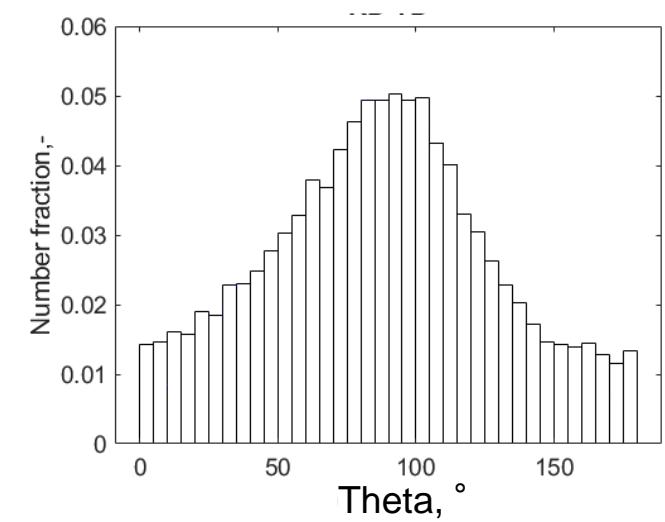
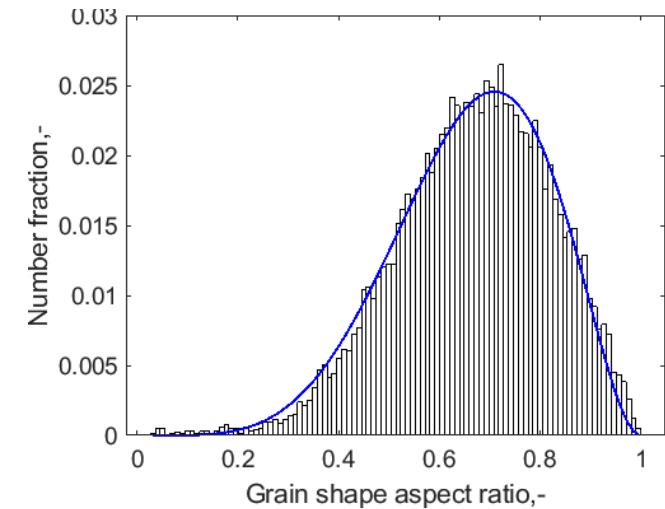
The angle between the major axis direction of a fitted ellipse the horizontal direction.

Beta distribution

$$f(x; \alpha, \beta) = \frac{1}{B(\alpha, \beta)} x^{\alpha-1} (1-x)^{\beta-1}$$

$$B(\alpha, \beta) = \int_0^1 x^{\alpha-1} (1-x)^{\beta-1} dt$$

$$\text{mean}(x) = \frac{1}{1 + \frac{\beta}{\alpha}}$$



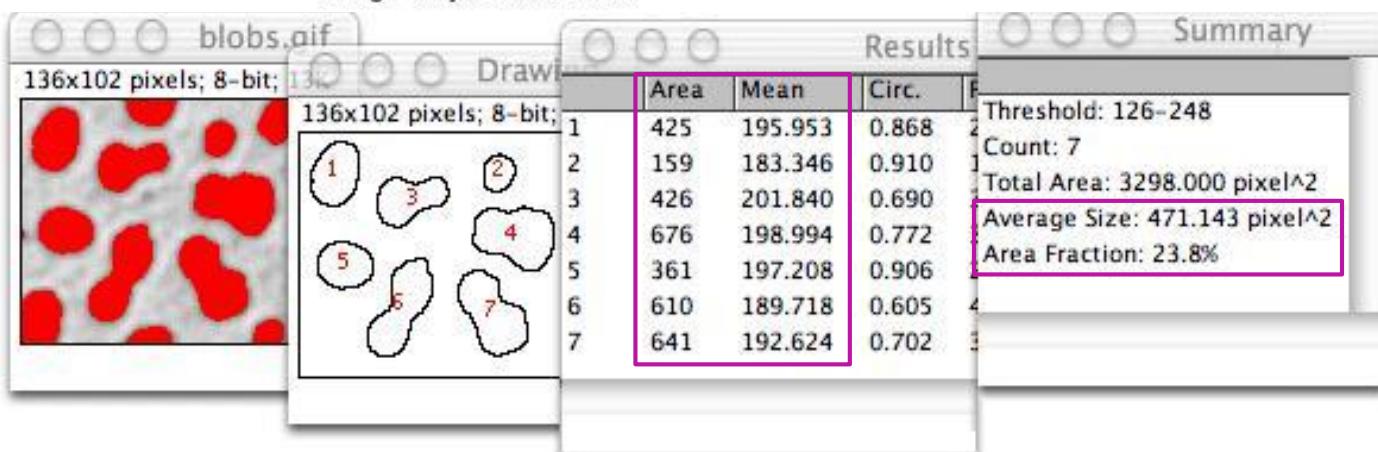
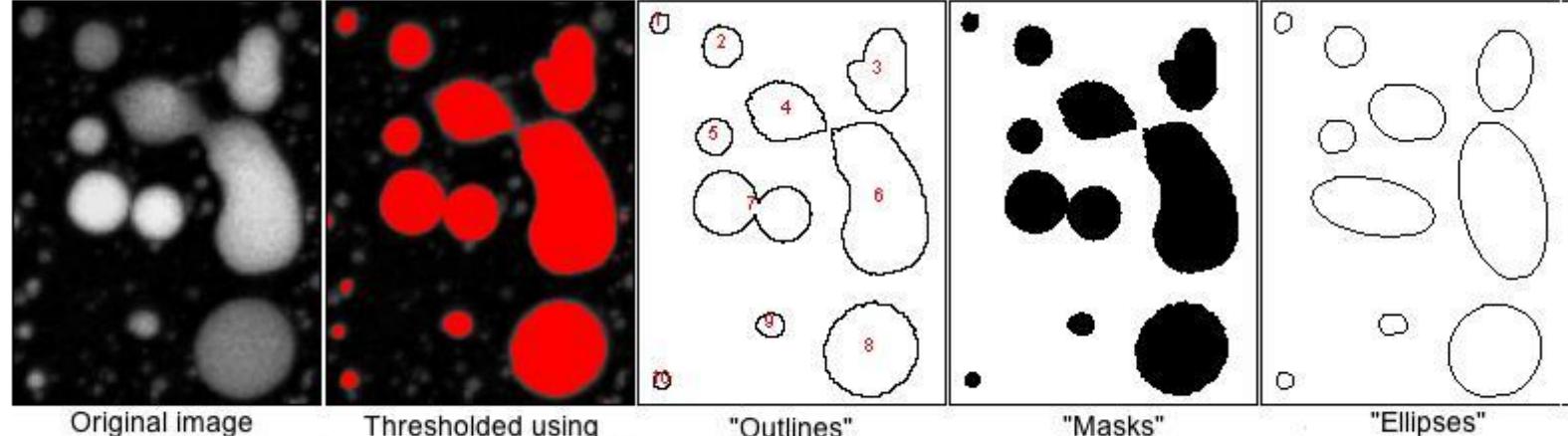
Software introduction

Software introduction

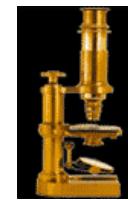
ImageJ is a Java-based image processing program developed at the National Institutes of Health and the Laboratory for Optical and Computational Instrumentation (LOCI, University of Wisconsin).

- Calculate area and pixel value statistics of user-defined selections and intensity-thresholded objects.
- Measure distances and angles.
- Create density histograms and line profile plots.
- etc.

ImageJ for particle size analysis



<https://imagej.nih.gov/ij/docs/menus/analyze.html>



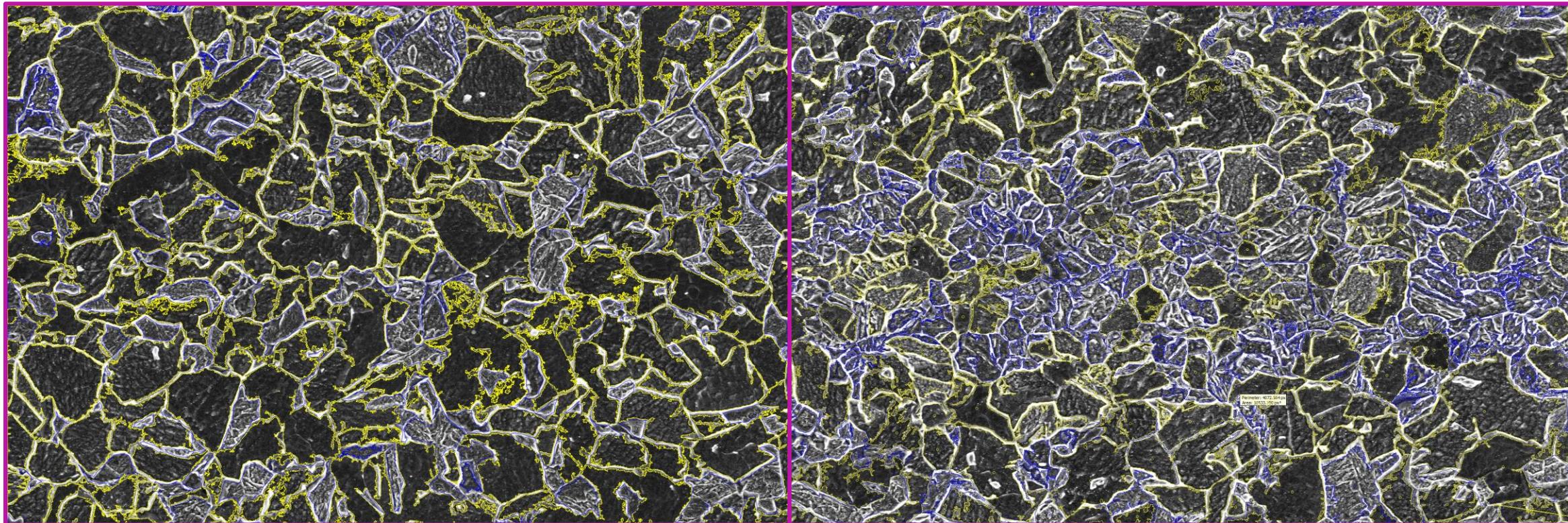
Software introduction

ImageJ for phase fraction analysis



ImageJ

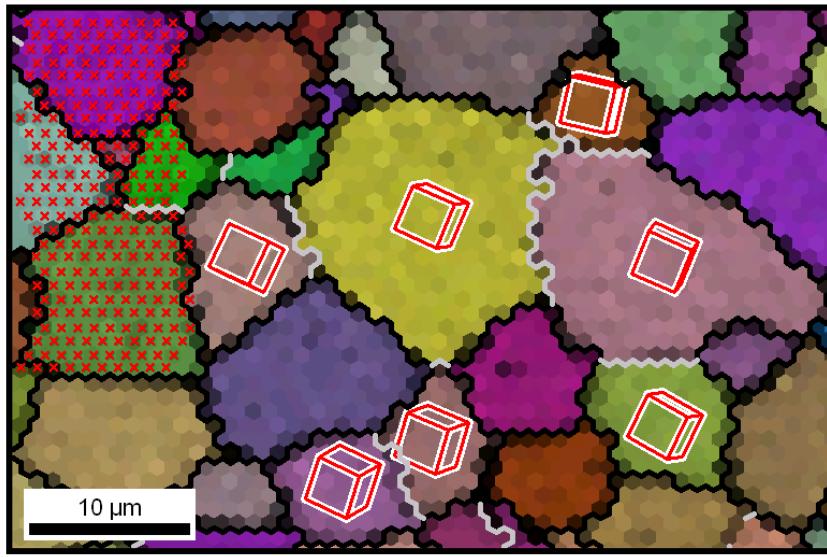
Image Processing & Analysis in Java



<https://www.sciencedirect.com/science/article/pii/S0749641919303729>

Software introduction

EBSD data

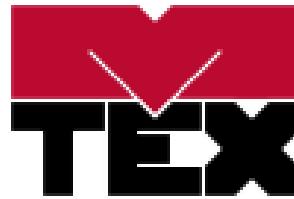


S. Zaefferer (MPIE): Microstructures, Microscopy & Modelling.

- phi1, PHI, phi2/Euler123: Euler angles
- XY: position, coordinates
- Phase/Grain: phase/grain ID
- Image Quality(IQ)/ Band Contrast(BC): intensity of the Kikuchi bands
- Confidence Index(CI)/Mean Angular Deviation (MAD): accuracy of calibrated Kikuchi pattern



Matlab MTEX toolbox



.ang type data

```
# Column 1-3: phil, PHI, phi2 (orientation of point in radians)
# Column 4-5: x, y (coordinates of point in microns)
# Column 6: IQ (image quality)
# Column 7: CI (confidence index)
# Column 8: Fit (degrees)
# Column 9: Grain ID (integer)
# Column 10: edge (1 for grains at edges of scan and 0 for interior grains)
# Column 11: phase name
```

3.25335	1.86772	1.01598	0.00000	0.00000	5619.8	0.641	0.87	1 1 Iron - Alpha
3.25490	1.86773	1.01857	5.00000	0.00000	5678.2	0.632	0.82	1 1 Iron - Alpha
3.25762	1.86577	1.01771	10.00000	0.00000	5724.1	0.659	0.90	1 1 Iron - Alpha
3.25587	1.86793	1.01924	15.00000	0.00000	5741.8	0.659	0.85	1 1 Iron - Alpha
3.25937	1.87042	1.01489	20.00000	0.00000	5860.7	0.659	0.87	1 1 Iron - Alpha
3.25567	1.86302	1.01263	25.00000	0.00000	5520.4	0.659	0.90	1 1 Iron - Alpha
1.01877	1.90168	4.30771	30.00000	0.00000	4587.9	0.627	1.02	2 1 Iron - Alpha
5.86586	1.35337	35.00000	0.00000	0.00000	4826.1	0.950	0.71	2 1 Iron - Alpha
5.86674	1.10970	35.414	40.00000	0.00000	4770.1	0.932	0.67	2 1 Iron - Alpha
5.86992	1.18950	4.34919	45.00000	0.00000	4928.0	0.955	0.71	2 1 Iron - Alpha
2.21082	0.21942	5.04289	50.00000	0.00000	4518.8	0.500	1.06	3 1 Iron - Alpha
4.10639	1.50440	4.92565	55.00000	0.00000	4675.3	0.500	0.93	3 1 Iron - Alpha

in radian

.ctf type data

Phase	X	Y	Bands	Error	Euler1	Euler2	Euler3	MAD	BC	BS	
5	0.0000	0.0000	11	0	314.5405	30.5703	80.0588	0.7228	173	96	
5	1.0000	0.0000	12	0	314.0926	30.7198	80.5870	0.4773	180	111	
5	2.0000	0.0000	12	0	314.0436	30.7488	80.7497	0.4559	186	121	
5	3.0000	0.0000	12	0	313.9620	30.7578	80.7800	0.5206	187	122	
5	4.0000	0.0000	12	0	314.2286	30.9458	80.4102	0.4530	188	126	
5	5.0000	0.0000	12	0	314.1919	5262	0.4533	190	124		
5	6.0000	0.0000	12	0	313.7010	0110	0.5802	172	111		
e	7.0000	0.0000	9	0	301.1472	34.0793	40.0444	0.3107	195	100	

in degree



Software introduction

Access

Matlab: Aalto software download service Download.aalto.fi (Aalto user credentials log in is needed).

MTEX toolbox: open-sourced software <https://mtex-toolbox.github.io/>

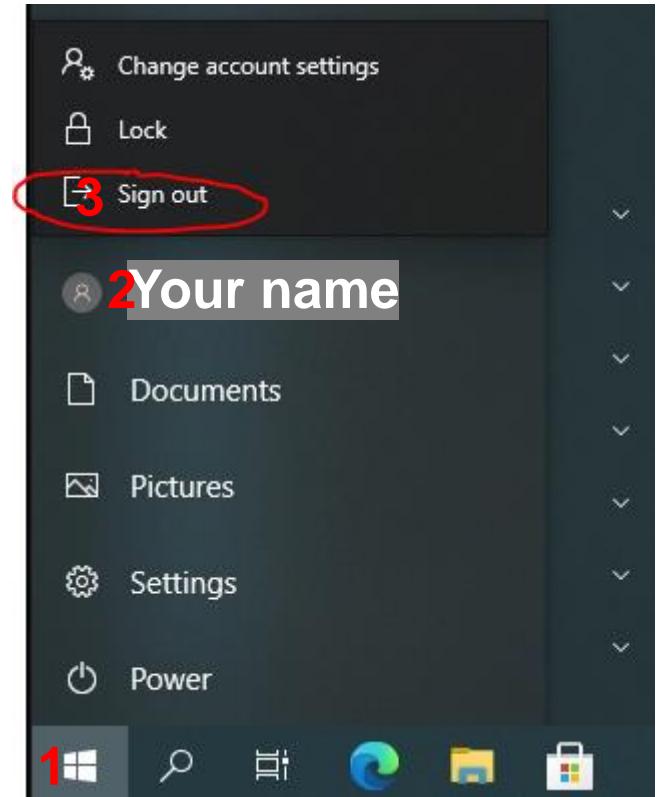
Aalto VDI system: mfavdi.aalto.fi, or VMware Horizon Client vdi.aalto.fi, for more information, please refer to [Remote access to Windows classroom computers](#).

IMPORTANT! Please remember to do '**Sign Out**' after the session (NOT Disconnect). Click your username in Start and click 'Sign Out'.

Basic Rule: Please use POINT as the decimal separator, NO COMMA!

Please download and extract the zip file of MTEX 5.4.0., and copy it to your Matlab work direction or add it to the Matlab work path.

Please also download the 'EBSDdata.ctf' file as input.



Mtex toolbox installation

[News](#) [Download](#) [People](#) [Documentation](#) [Resources](#)

search...

Download

Installation

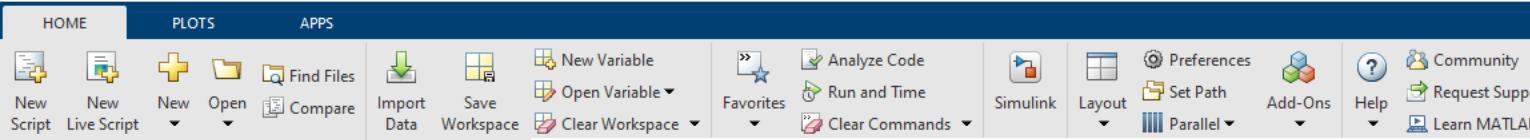
1. download and extract the zip file to an arbitrary folder
2. start Matlab (version 2016b or newer required - older versions have not been tested)
3. change the current folder in Matlab to the folder where MTEX is installed
4. type `startup_mtex` into the command window
5. click one of the menu items to import data or to consult the documentation

In case you experience any problems, especially on Mac OSX, have a look at our [trouble shooting page](#)

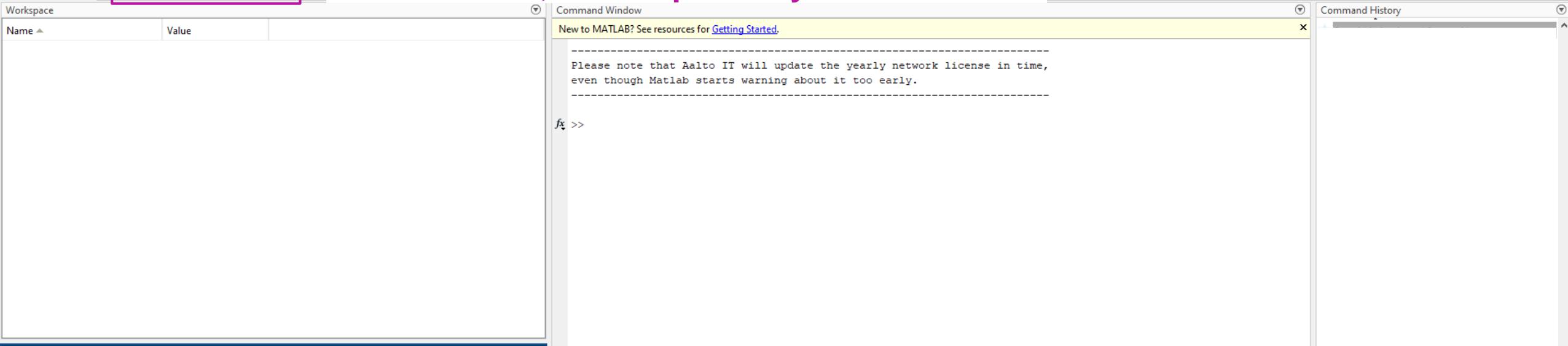
Downloads

File Name	Release Date	Comments	Downloads
mtex-5.5.beta.3.zip	October 2020	low angle grain boundary analysis, orientation embeddings, changelog	275
mtex-5.4.0.zip	July 2020	parent grain reconstruction, changelog	2310
mtex-5.3.1.zip	June 2020	bug fixes, boundary curvature, changelog	1034
mtex-5.2.8.zip	January 2020	plasticity tensors, changelog	2899
mtex-5.1.1.zip	June 2018	GND calculation and birefringence, changelog	8291
MTEXGUI-2.4.zip	Sep 2018	graphical user interface for analyzing EBSD data by J. Hiscock	
mtex-5.0.3.zip	March 2018	Crystal Shapes and spherical functions, changelog	2796
MTEXannotateGUI.zip	February 2018	graphical user interface for analyzing EBSD data by J. Hiscock	
mtex-4.5.2.zip	November 2017	3d orientation and ODF plots, changelog	4420
mtex-4.4.0.zip	January 2017	Slip Systems, Taylor calculation, changelog	1252
mtex-4.3.2.zip	July 2016	GND calculation, changelog	1662

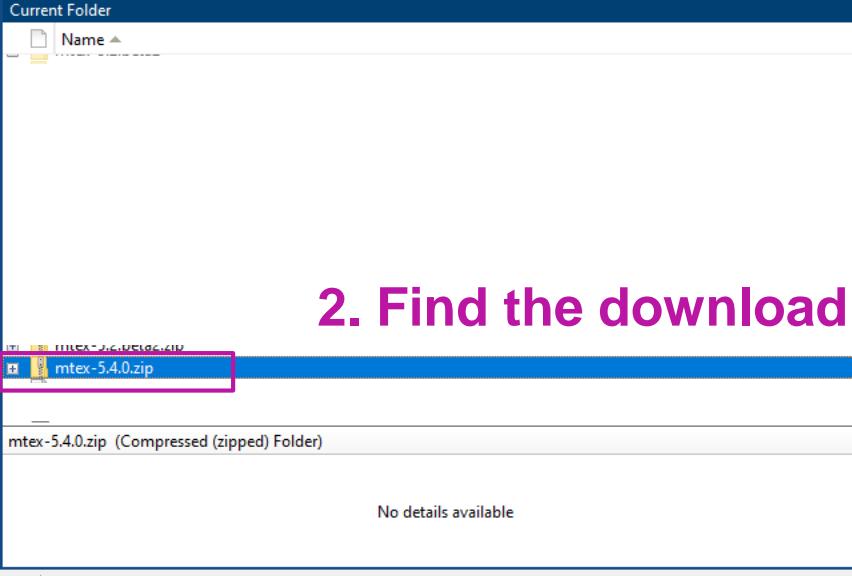
Download 'mtex-5.4.0.zip' to your Matlab folder



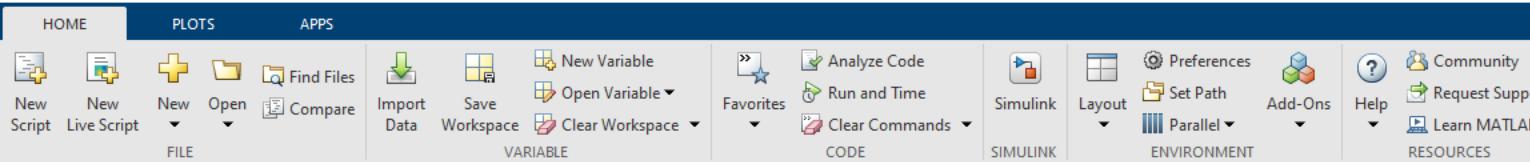
1. Run Matlab, set the path to your work folder



2. Find the download .zip file. Double click to extract



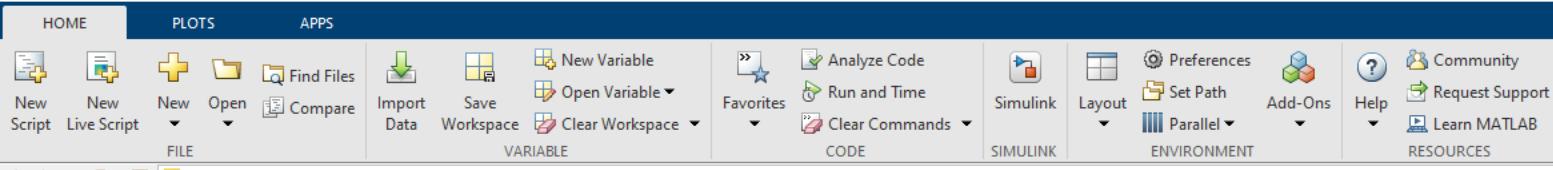
Mtex installation



Mtex installation

The screenshot shows the MATLAB desktop environment. The 'Current Folder' browser on the left displays two folders: 'mTEX-5.2.beta2' and 'mTEX-5.4.0'. A context menu is open over the 'mTEX-5.4.0' folder, with a message box overlaid stating: 'This folder is not on your MATLAB path. Double-click mTEX-5.4.0 to make it your current folder or select "Add to Path" from its context menu to add it to your path.' There is a 'Do not show this message again' button at the bottom of the message box. The 'Command Window' in the center contains the text: 'Please note that Aalto IT will update the yearly network license in time, even though Matlab starts warning about it too early.' The 'Command History' window on the right is currently empty.

Double click the extracted folder to add it to the work path



Mtex installation

The screenshot shows the MATLAB R2020a interface with the following components:

- Current Folder Browser:** Located at the bottom left, it shows the directory structure of the MTEX distribution. The file `startup.m` is selected and has a context menu open.
- Context Menu for `startup.m`:** The menu items include Open, Open as Live Function, Hide Details, Run (which is highlighted with a pink rectangle), Run Script as Batch Job, View Help, Show in Explorer, Create Zip File, Rename, Delete, Compare Selected Files/Folders, Compare Against, Cut, Copy, Paste, and Indicate Files Not on Path.
- Command Window:** Located in the center, it displays a message about network license updates and the MATLAB startup prompt (`>>`).
- Workspace Browser:** Located at the top left, it shows the current variables in the workspace.
- Command History:** Located at the top right, it shows a history of previously run commands.

A large pink text overlay in the center of the image reads: "Right click 'startup.m' function and click 'Run' to install MTEX".

Mtex installation

Generated data will be shown in the workspace window

Check commands in the command history window

Type commands in the command window

The screenshot shows the MATLAB R2020a interface with the following components visible:

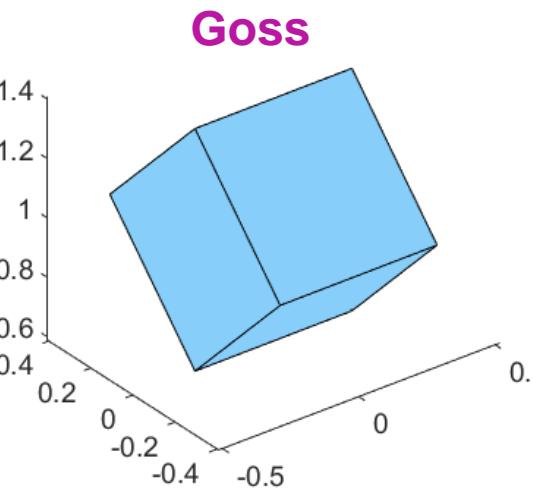
- HOME**, **PLOTS**, **APPS** tabs at the top.
- FILE** tab selected, showing options: New Script, New Live Script, New, Open, Compare, Import Data, Save Workspace, New Variable, Open Variable, Favorites, Analyze Code, Run and Time, Clear Commands, Simulink, Preferences, Set Path, Add-Ons, Parallel, Help, Community, Request Support, Learn MATLAB.
- VARIABLE** tab selected under FILE.
- Workspace** window showing a table with columns "Name" and "Value".
- Current Folder** browser showing the directory structure of the installed MTEX 5.4.0 package, including files like tests, tools, CHANGELOG, compile-mtex, COPYING.txt, info.xml, install_mtex.m, Makefile, mtex_icon.gif, mtex_settings.m, mtex-include, README.md, startup.m, startup_mtex.m, and uninstall_mtex.m.
- Command Window** showing the output of the startup command:

```
Please note that Aalto IT will update the yearly network license in time,  
even though Matlab starts warning about it too early.  
  
-> startup  
initialize MTEX 5.4.0 .... done!  
  
MTEX 5.4.0 (show documentation)  
Import pole figure data  
Import EBSD data  
Import ODF data  
  
Uninstall MTEX
```
- Command History** window on the right.

Mtex - Crystal orientation operation

Commands for crystal orientation

```
% Define the crystal structure, e.g. for bcc/fcc:  
cs = crystalSymmetry('m-3m')  
ss = specimenSymmetry('mmm')  
  
%% Define the crystal orientation  
% Goss orientation (0°, 45°, 0°), (011)[100]  
% ori = orientation.byEuler(phi1,Phi,phi2,CS,SS)  
% ori = orientation.byMiller([h k l],[u v w],CS)  
oriGoss1 = orientation.byEuler(0*degree,45*degree,0*degree,cs,ss)  
oriGoss2 = orientation.byMiller([0 1 1],[1 0 0],cs,ss)  
  
%% Calculate the orientation matrix  
GmatrixGoss1 = oriGoss1.matrix'  
GmatrixGoss2 = oriGoss2.matrix'  
  
%% Plot the rotate cube for the orientation  
cS = crystalShape.cube(cs)  
figure  
plot([0,0]+oriGoss1 * cS)  
axis on
```



```
oriGoss2 = orientation (show methods, plot)  
size: 1 x 1  
crystal symmetry : m-3m  
specimen symmetry: mmm  
  
Bunge Euler angles in degree  
phil Phi phi2 Inv.  
0 45 0 0
```

Euler angles

Orientation matrix

```
GmatrixGoss1 =  
  
1.0000 0.0000 0.0000  
-0.0000 0.7071 0.7071  
0.0000 -0.7071 0.7071
```

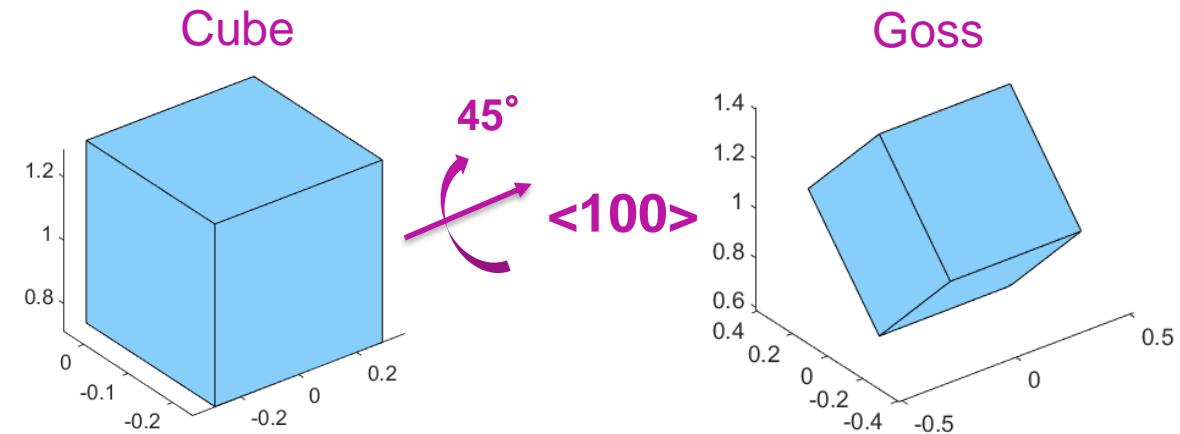
```
GmatrixGoss2 =  
  
1.0000 0.0000 -0.0000  
0.0000 0.7071 0.7071  
0.0000 -0.7071 0.7071
```

Commands for crystal orientation

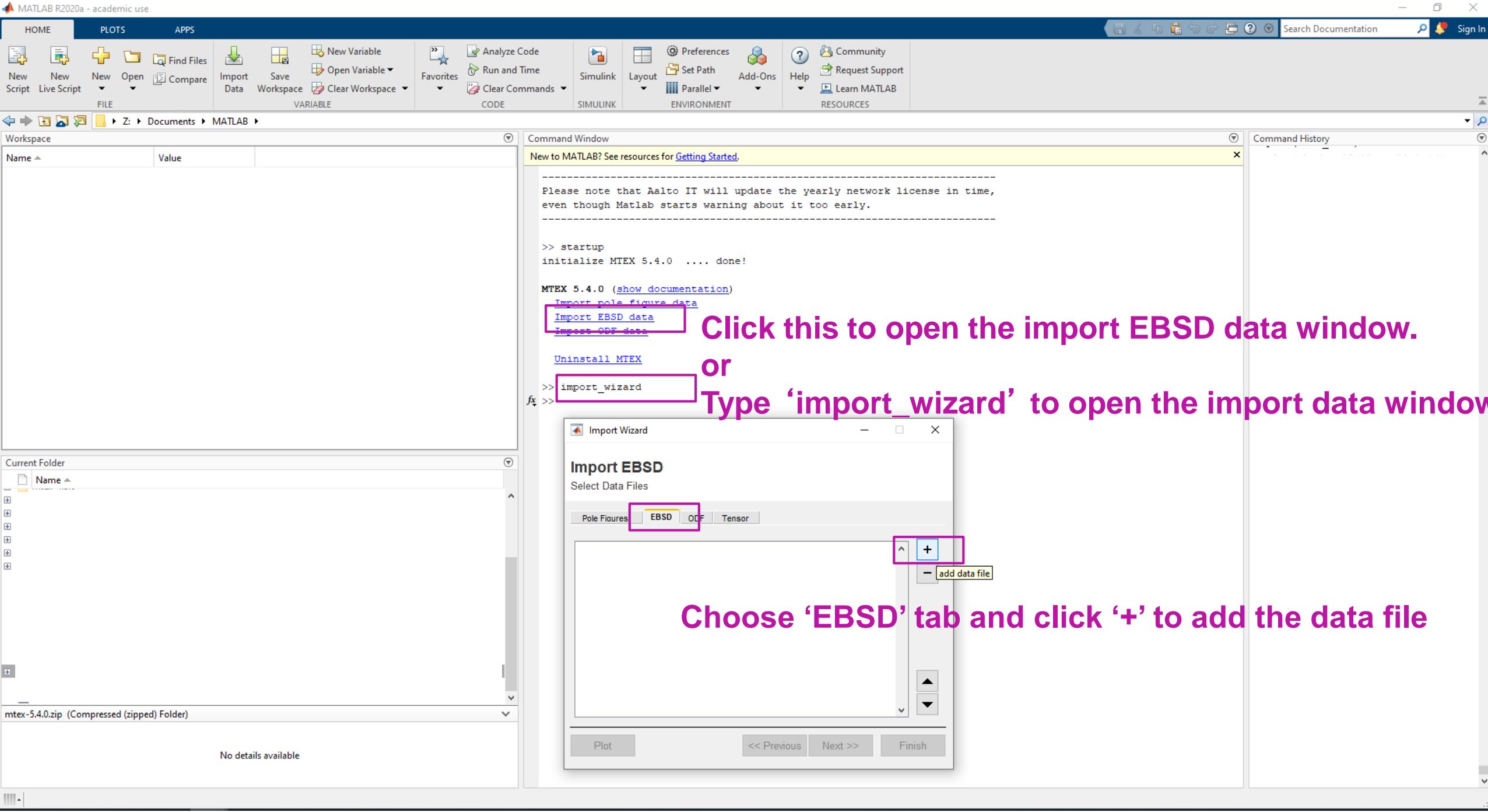
```
% Define another crystal orientation and plot, e.g. Cube orientation (90° ,0° ,0° ):  
oriCube1 = orientation.byEuler(90*degree,0*degree,0*degree,cs,ss)  
figure  
plot([0,0]+oriCube * cS)  
axis on  
%% Calculate the rotation Axis/Angle  
uvw = axis(oriGoss1,oriCube)  
theta = angle(oriGoss1,oriCube)/degree
```

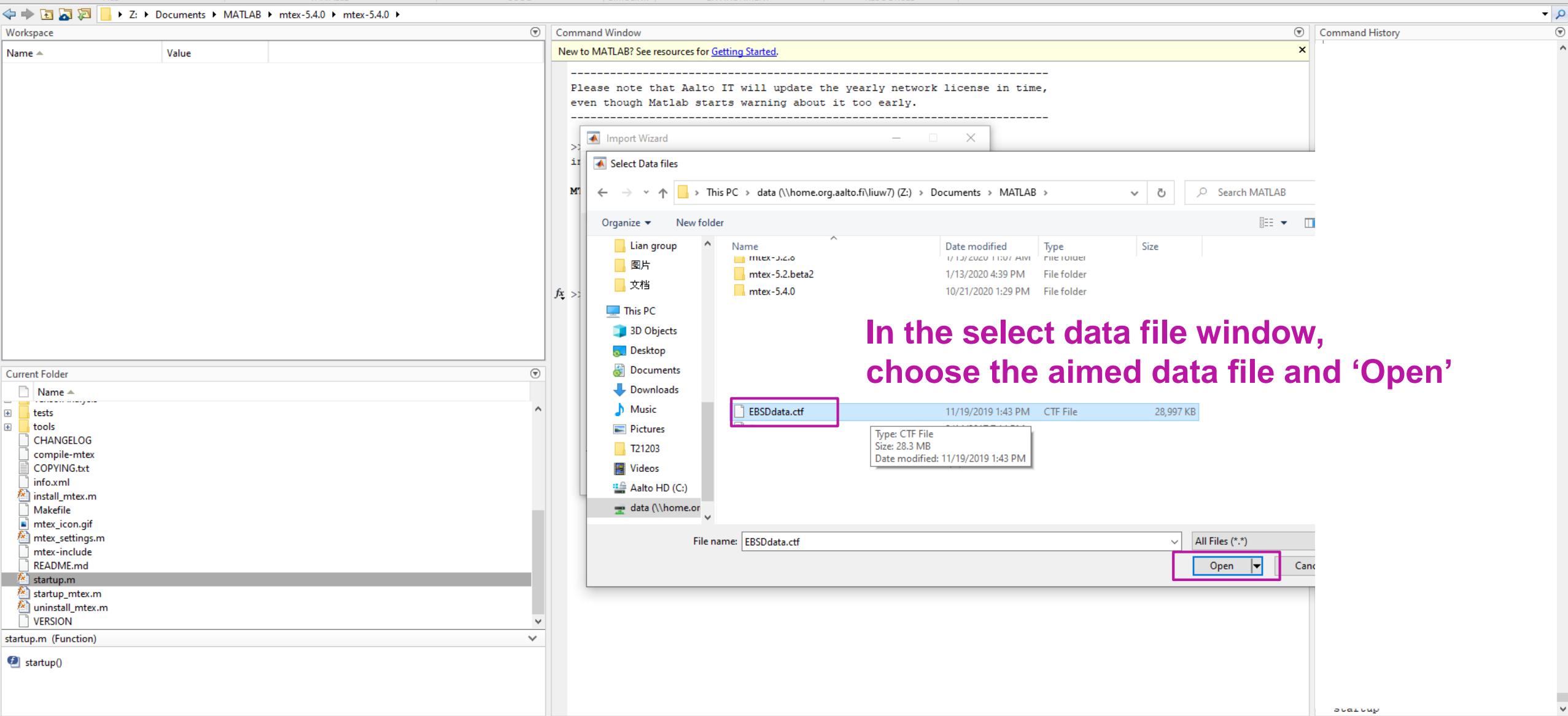
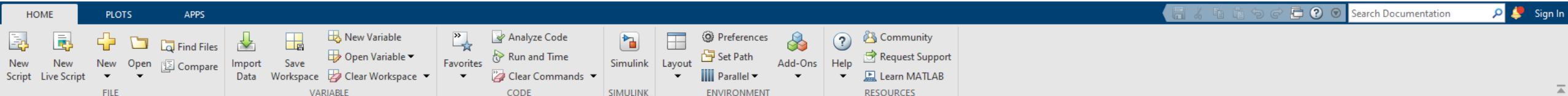
```
>> uvw = axis(oriGoss1,oriCube)  
theta = angle(oriGoss1,oriCube)/degree  
  
uvw = vector3d (show methods, plot)  
size: 1 x 1  
x y z  
1 0 0  
common rotation axis  
  
theta =
```

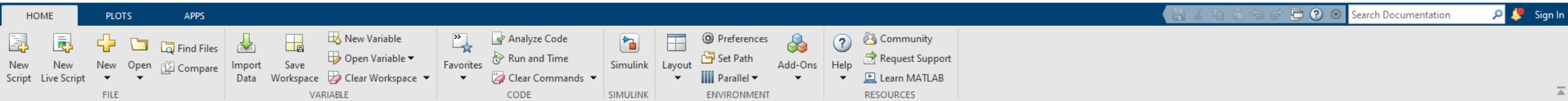
45.0000 rotation angle -> misorientation



Mtex - EBSD data analysis







Z: > Documents > MATLAB > mtex-5.4.0 > mtex-5.4.0 >

Workspace

Name	Value

Command Window

New to MATLAB? See resources for [Getting Started](#).

Please note that Aalto IT will update the yearly network license in time, even though Matlab starts warning about it too early.

Import Wizard

Import EBSD

Select Data Files

Pole Figures EBSD ODF Tensor

EBSDdata.ctf

Plot << Previous Next >> Finish

Current Folder

- Name
- tests
- tools
- CHANGELOG
- compile-mtex
- COPYING.txt
- info.xml
- install_mtex.m
- Makefile
- mtex_icon.gif
- mtex_settings.m
- mtex-include
- README.md
- startup.m
- startup_mtex.m
- uninstall_mtex.m
- VERSION

startup.m (Function)

startup()

After add the EBSD data file, click for 'Next' step

Crystal Reference Frame for Phase 0

Crystal Symmetry

Mineral Indexed Not Indexed
mineral name notIndexed
plotting color

Crystal Coordinate System

Point Group 1
Axis Length a , b , c
Axis Angle alpha , beta , gamma

Crystal Reference Frame for Phase 1

Crystal Symmetry

Mineral Indexed Not Indexed
mineral name Ti3Al - alpha2
plotting color

Crystal Coordinate System

Point Group 6/mmm
Axis Length a , b , c
Axis Angle alpha , beta , gamma

Crystal Reference Frame for Phase 2

Crystal Symmetry

Mineral Indexed Not Indexed
mineral name TiAl - gamma
plotting color

Crystal Coordinate System

Point Group 4/mmm
Axis Length a , b , c
Axis Angle alpha , beta , gamma

Different Phase parameter/setting will be shown based on the measured data. Just leave with the default and click 'Next'.

Phase 0

Crystal Syn

Mineral Indexed Not Indexed
mineral name Titanium-Cubic
plotting color

Crystal Coordinate System

Point Group m-3m
Axis Length a , b , c
Axis Angle alpha , beta , gamma

Phase 1

Crystal Syn

Mineral Indexed Not Indexed
mineral name Aluminium
plotting color

Crystal Coordinate System

Point Group m-3m
Axis Length a , b , c
Axis Angle alpha , beta , gamma

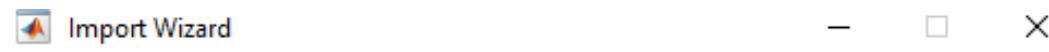
Phase 2

Crystal Syn

Mineral Indexed Not Indexed
mineral name Aluminium
plotting color

Crystal Coordinate System

Point Group m-3m
Axis Length a , b , c
Axis Angle alpha , beta , gamma



Specimen Reference Frame

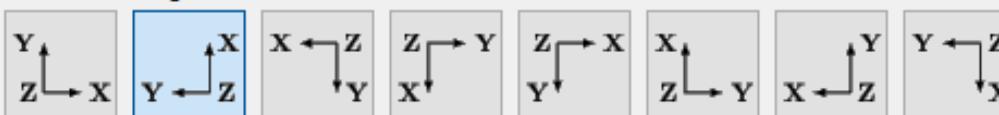
Specimen Symmetry

Specimen Coordinate System

rotate data by Euler angles (Bunge) in degree

apply rotation to Euler angles and spatial coordinates
 apply rotation only to Euler angles
 apply rotation only to spatial coordinates
 use CTF interface flag 'convertSpatial2EulerReferenceFrame'
 use CTF interface flag 'convertEuler2SpatialReferenceFrame'

MTEX Plotting Convention



Plot the data to verify that the coordinate system is properly aligned!

Import Data

Select Method

Summary of EBSD data to be imported:

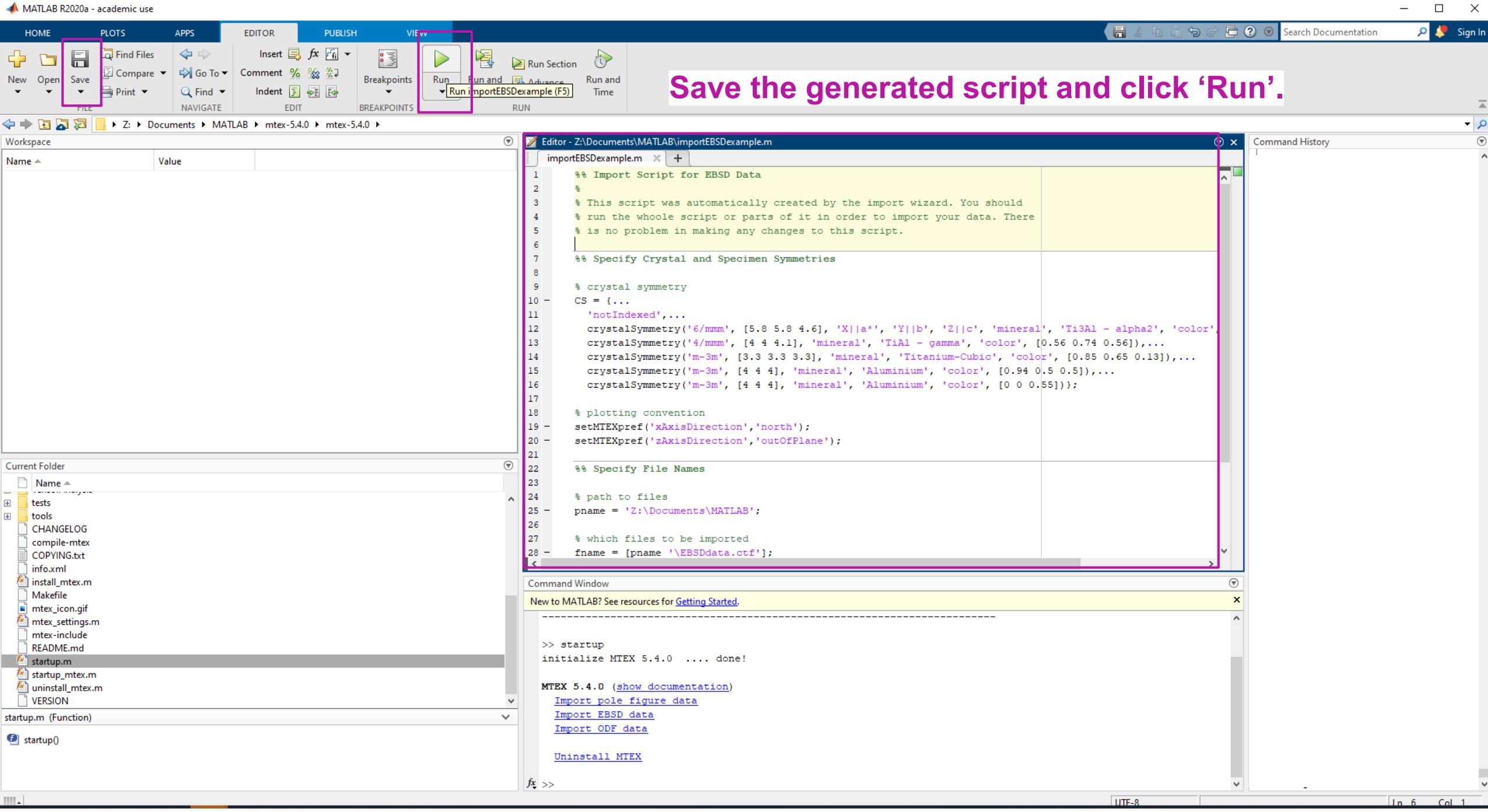
```
phase 0 (not Indexed): notIndexed, 22330 orientations
phase 1 (Ti3Al - alpha2): symmetry 6/mmm, 0 orientations
phase 2 (TiAl - gamma): symmetry 4/mmm, 0 orientations
phase 3 (Titanium-Cubic): symmetry m-3m, 0 orientations
phase 4 (Aluminium): symmetry m-3m, 0 orientations
phase 5 (Aluminium): symmetry m-3m, 438767 orientations
```

Import to

script (m-file) workspace variable

Specimen coordinate setting,
use the default.

Generate the script file for importing.



Commands for importing EBSD data - Examples

```
%% Specify Crystal and Specimen Symmetries
% crystal symmetry
CS = {...
    'notIndexed',...
    crystalSymmetry('6/mmm', [5.8 5.8 4.6], 'X||a*', 'Y||b', 'Z||c', 'mineral', 'Ti3Al - alpha2',...
    'color', [0.53 0.81 0.98]),...
    crystalSymmetry('4/mmm', [4 4 4.1], 'mineral', 'TiAl - gamma', 'color', [0.56 0.74 0.56]),...
    crystalSymmetry('m-3m', [3.3 3.3 3.3], 'mineral', 'Titanium-Cubic', 'color', [0.85 0.65
0.13]),...
    crystalSymmetry('m-3m', [4 4 4], 'mineral', 'Aluminium', 'color', [0.94 0.5 0.5]),...
    crystalSymmetry('m-3m', [4 4 4], 'mineral', 'Aluminium', 'color', [0 0 0.55]));
% plotting convention
setMTEXpref('xAxisDirection','north');
setMTEXpref('zAxisDirection','outOfPlane');
```

```
%% Specify File Names
pname = 'Z:\Documents\MATLAB'; % path to files
fname = [pname '\EBSDdata.ctf']; % which files to be imported
```

```
%% Import the Data, create an EBSD variable containing the data
```

```
ebsd = EBSD.load(fname,CS,'interface','ctf','convertEuler2SpatialReferenceFrame');
```

1. Crystal structure setting for each phase,
default based on the measured EBDS data.

2. EBSD file path and name.

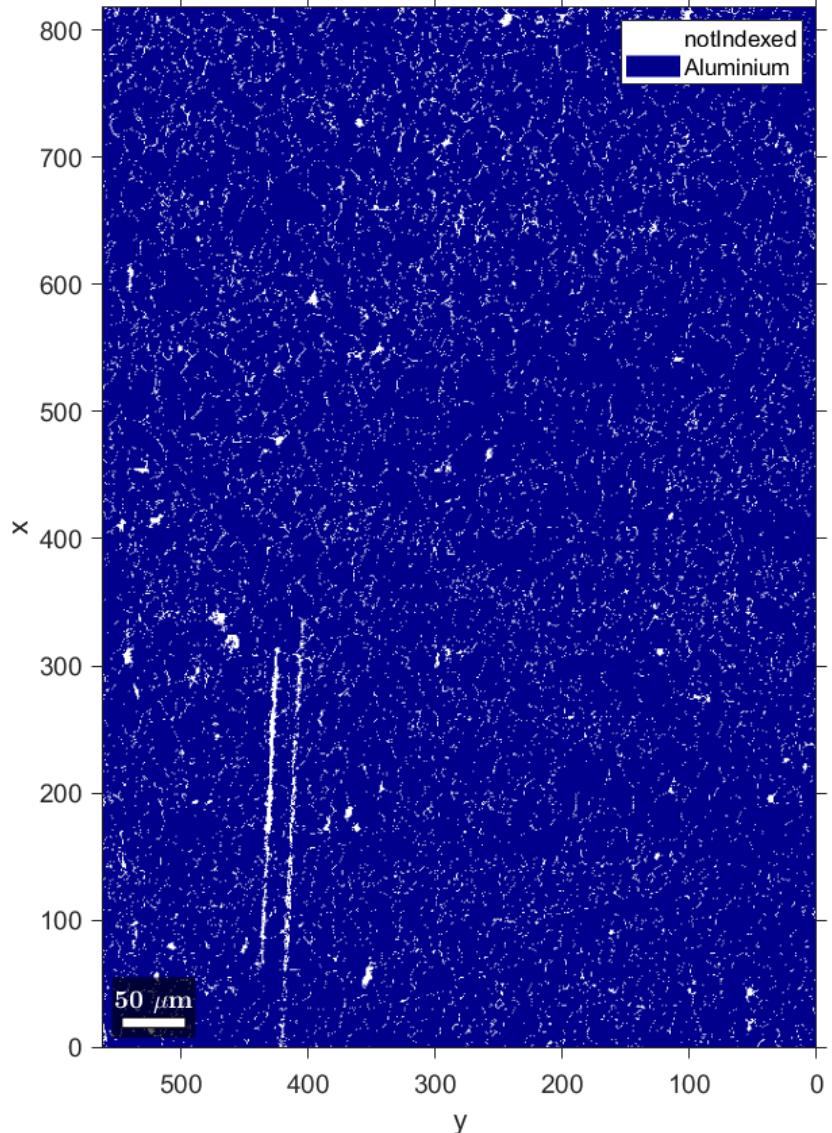
3. Load EBSD data.

Commands for EBSD data plotting

```
%% Initial analyses and plotting  
% Plot the Index/Phase map  
figure; plot(ebsd, 'coordinates', 'on');
```

Check the phase(s) in the investigated material and the measured area.

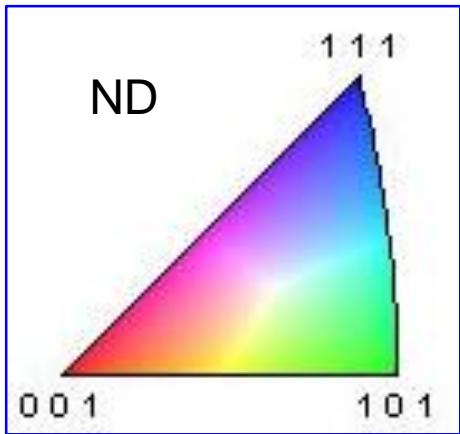
EBSD phase map + coordinates
Nonindexed
Aluminum



Commands for EBSD data plotting

```
% Plot the EBSD orientation map.
```

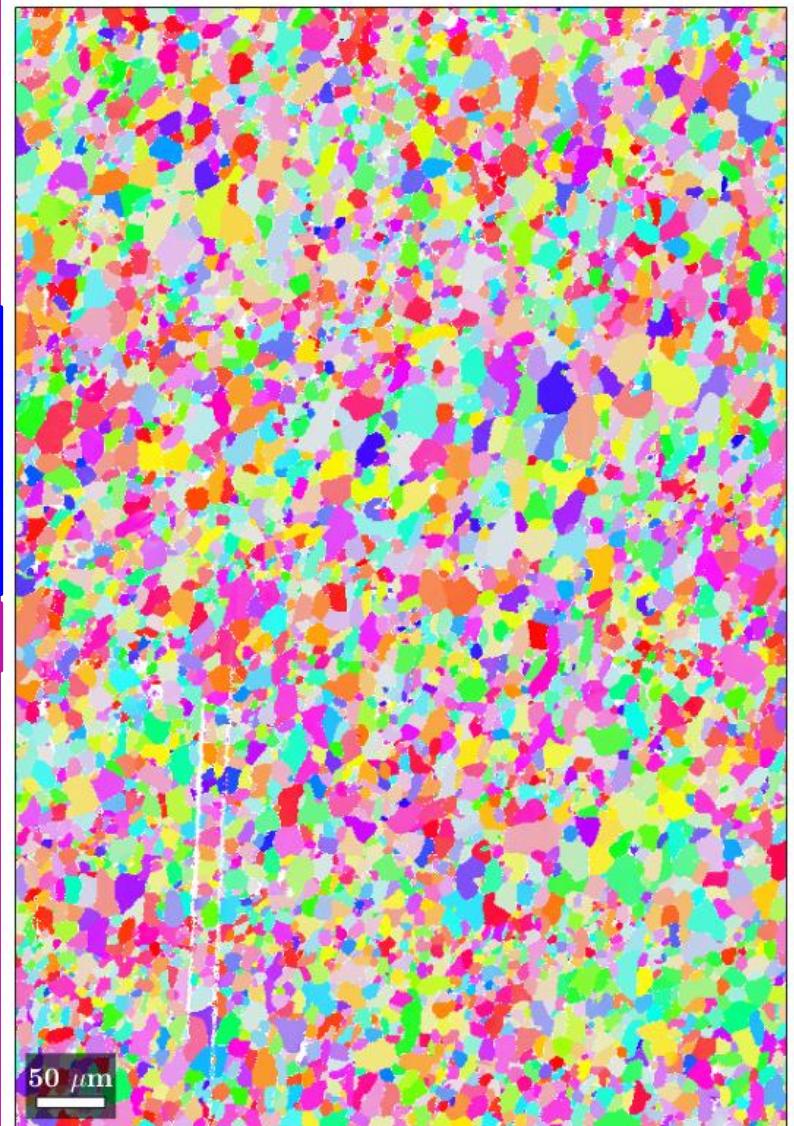
```
ipfKey = ipfColorKey(ebsd('Aluminium'));
ipfKey.inversePoleFigureDirection = vector3d.Z;
colors = ...
ipfKey.orientation2color...
(ebsd('Aluminium').orientations);
figure; plot(ebsd('Aluminium'), colors);
```



Plot the orientation map

Examples for the color map

EBSD orientation map for single phase



Commands for EBSD data analysis

```
%> Grain reconstruction  
% Consider only indexed & corrected data.  
ebsd_corrected = ebsd(ebsd.mad<1);  
ebsdcorri = ebsd_corrected('indexed');  
  
% Reconstruct the grain structure.  
[grains,ebsdcorri.grainId,ebsdcorri.mis2mean] = calcGrains(ebsdcorri,'angle',15*degree);  
initialGrainNr = length(grains);  
  
% Delete the very small grains which might be caused by the measurement error.  
ebsdcorri(grains(grains.grainSize<2)) = [];  
  
% Redo grain segmentation.  
[grains,ebsdcorri.grainId] = calcGrains(ebsdcorri,'angle',15*degree);  
  
% Pick up the focused phase.  
grainsAl=grains('Aluminium');  
totalGrainNr=length(grainsAl);  
  
% Plotting GrainOri maps  
figure; plot(grainsAl,grainsAl.meanOrientation);
```

Set the accuracy criterion for measured data, delete the points with $MAD>1$.

Pick up only indexed points.

Check the number of initial grains after first reconstruction.

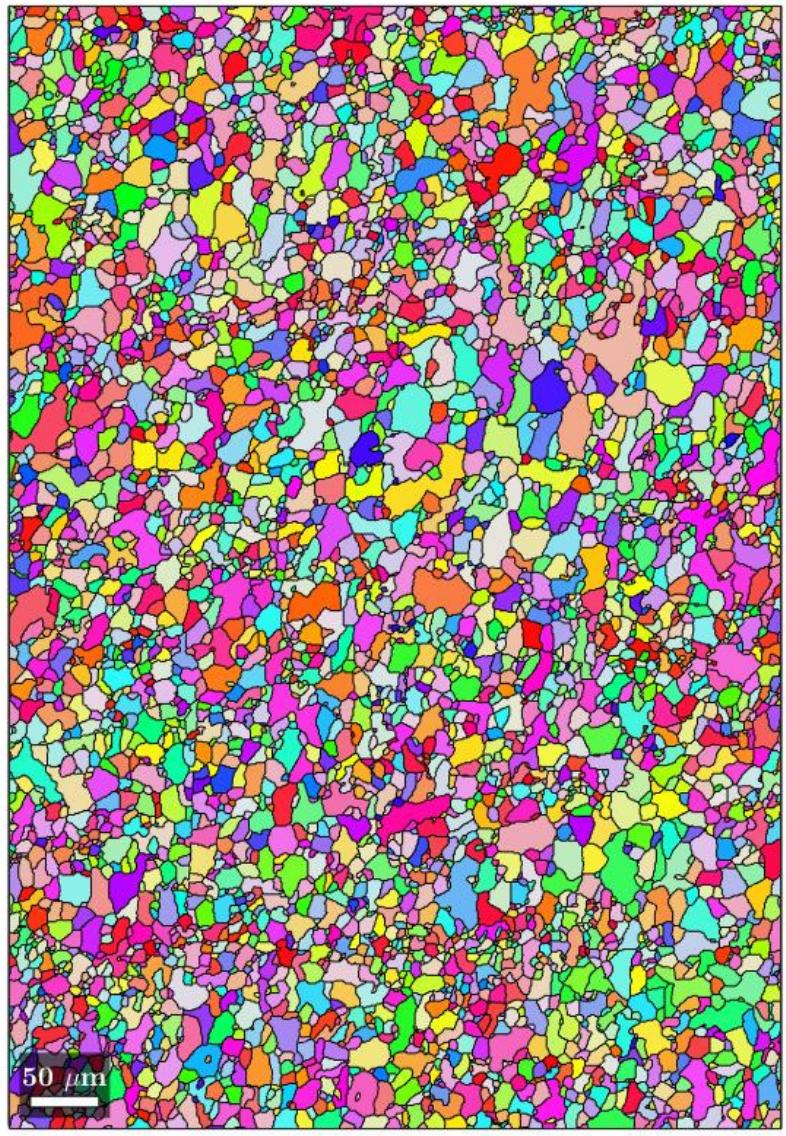
Set the grain boundary as 15° .

Delete grains with only 1 or 2 measured points.

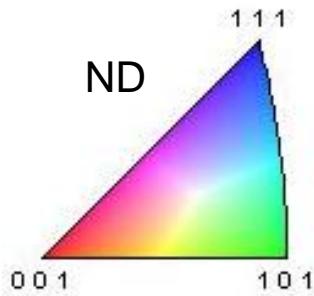
Pick up grains for the focused phase.

Check the number of total grains of the focused phase.

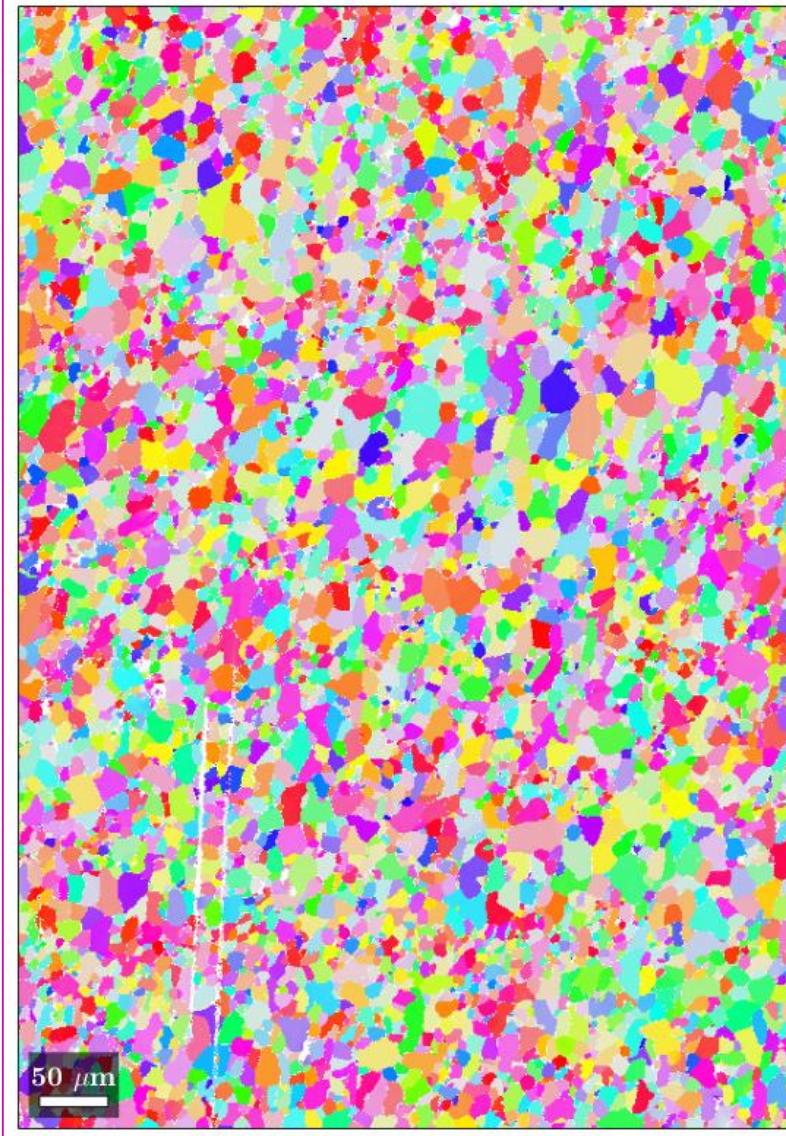
Plot grain map with grain mean orientation.



Grain mean
orientation map
with
grain boundaries
in black lines



EBSD
orientation map
for
scattered points



Commands for EBSD data analysis

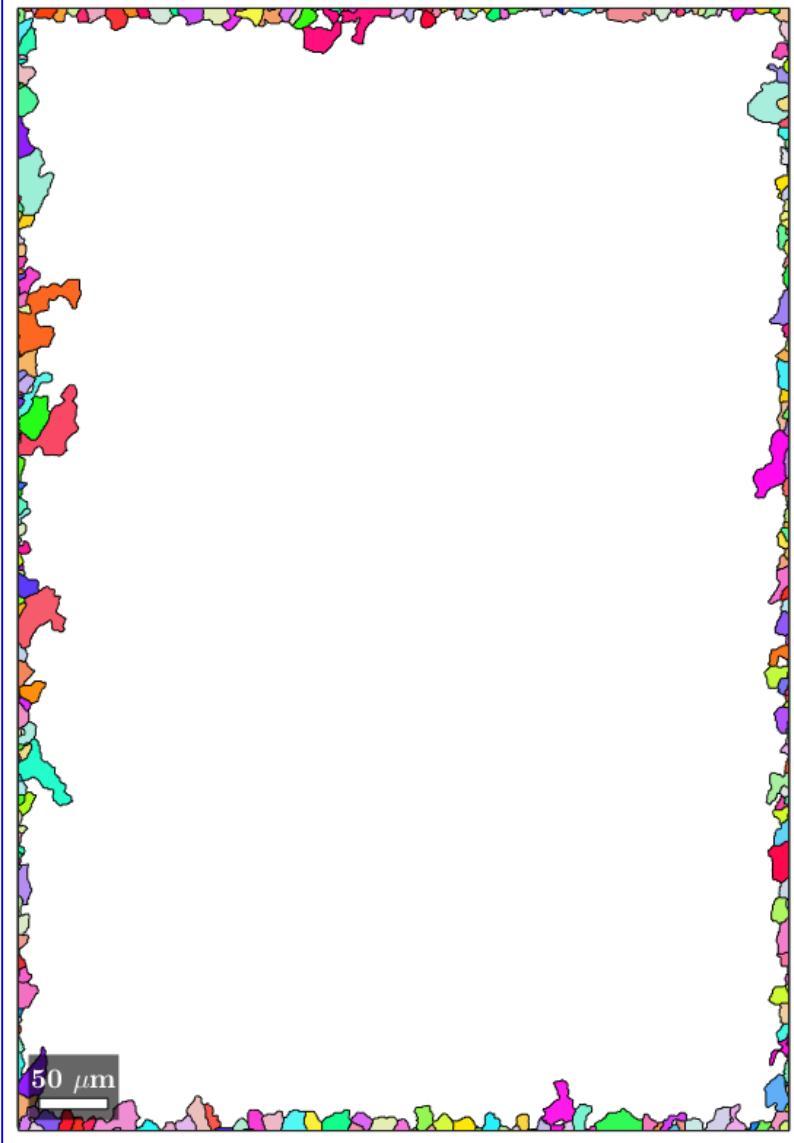
The grains at the measured area boundaries should be removed for grain size and shape analysis.

```
%% Grain Size & Shape Data analyses  
% Find the boundary grains.  
outerBoundary_id = any(grainsAl.boundary.grainId==0, 2);  
grain_id = grainsAl.boundary(outerBoundary_id).grainId;  
grain_id(grain_id==0) = [];  
  
% Plot the boundary grains with their mean orientations.  
figure; plot(grainsAl(grain_id),grainsAl(grain_id).meanOrientation);  
  
% Remove the boundary grains.  
grainsAl(grain_id) = [];  
innerGrainNr = length(grainsAl);  
  
% Plot the inner grains with their mean orientations.  
figure; plot(grainsAl,grainsAl.meanOrientation);
```

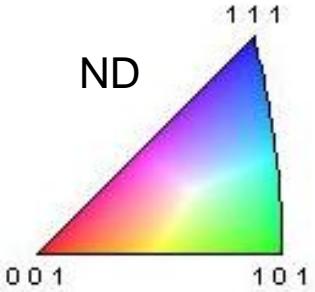
Plot the boundary grain map with grain mean orientation.

Check the number of inner grains of the focused phase.

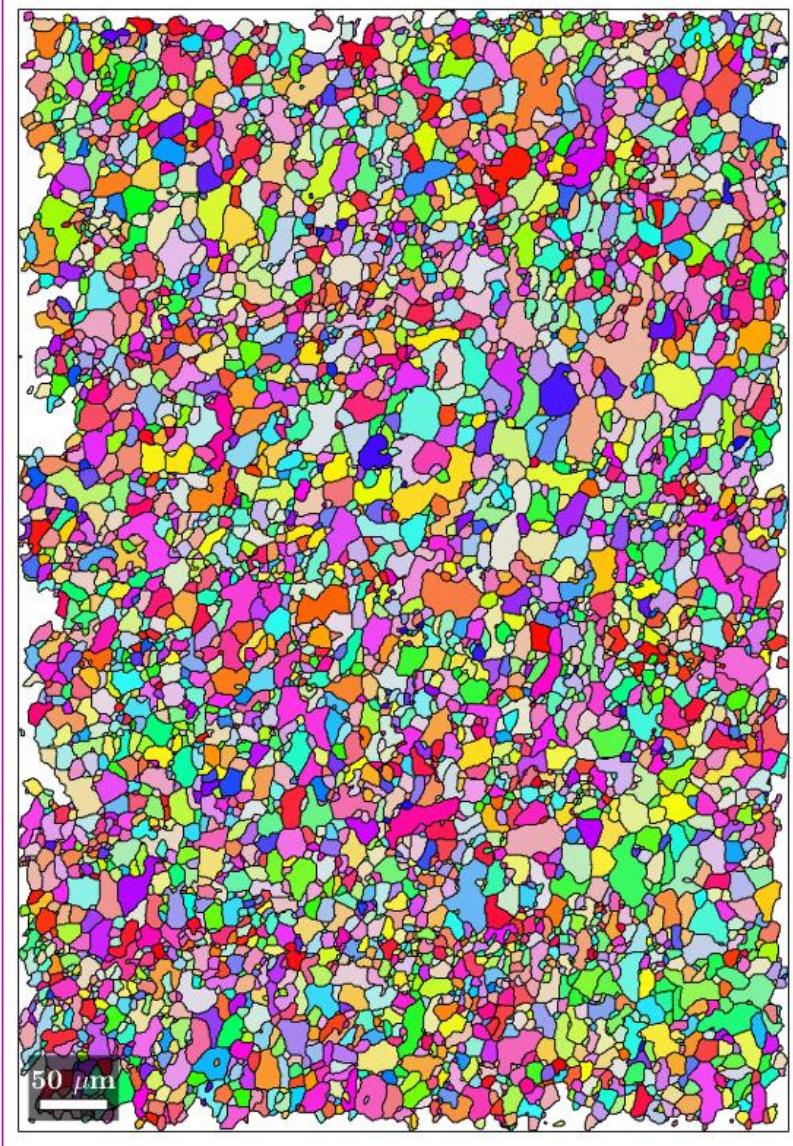
Plot the inner grain map with grain mean orientation.



Boundary grains
map with their
mean orientation



Inner grains map
with their mean
orientation
↓
For further grain
size and shape
analysis

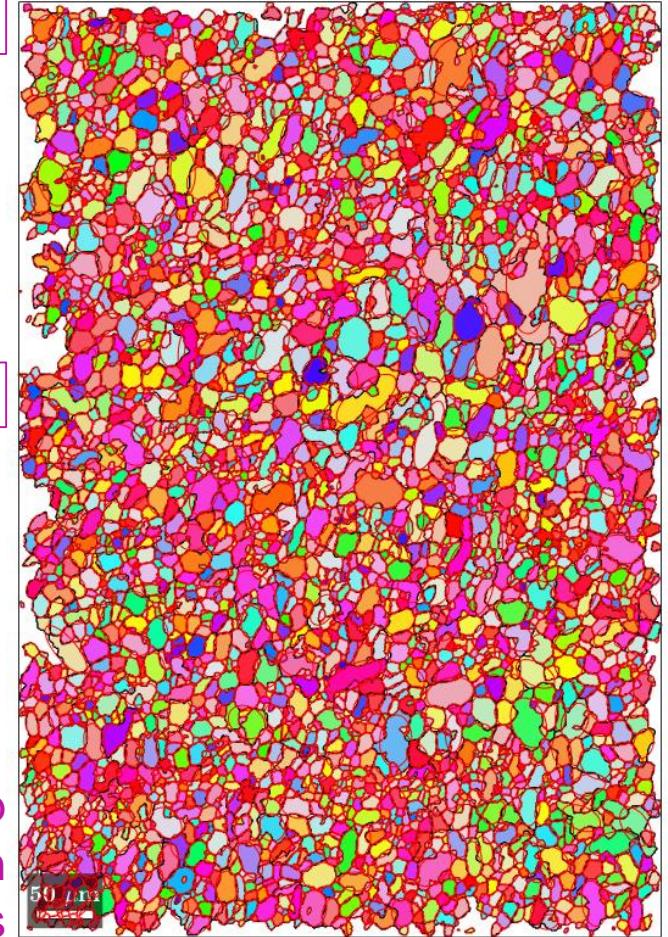


Commands for EBSD data analysis

```
% Extract grains data  
Grainarea = grainsAl.area; Grain area  
GraineqR = grainsAl.equivalentRadius; Equivalent grain radius  
GraineqD = GraineqR*2; Equivalent grain diameter  
Grainasp = 1./grainsAl.aspectRatio; Grain shape factor: aspect ratio
```

```
% Fit and plot the equivalent ellipses of grains  
[GrainfitEangle,GrainfitElongA,GrainfitEshortb] = fitEllipse(grainsAl);  
  
figure; Grain shape angle Ellipse grain longest and shortest axes.  
plot(grainsAl,grainsAl.meanOrientation,'linewidth',1);  
hold on;  
plotEllipse(grainsAl.centroid,GrainfitElongA,GrainfitEshortb, ...  
    GrainfitEangle,'lineColor','r');  
hold off;
```

Inner grains map
with mean orientation
and fitted equivalent ellipses



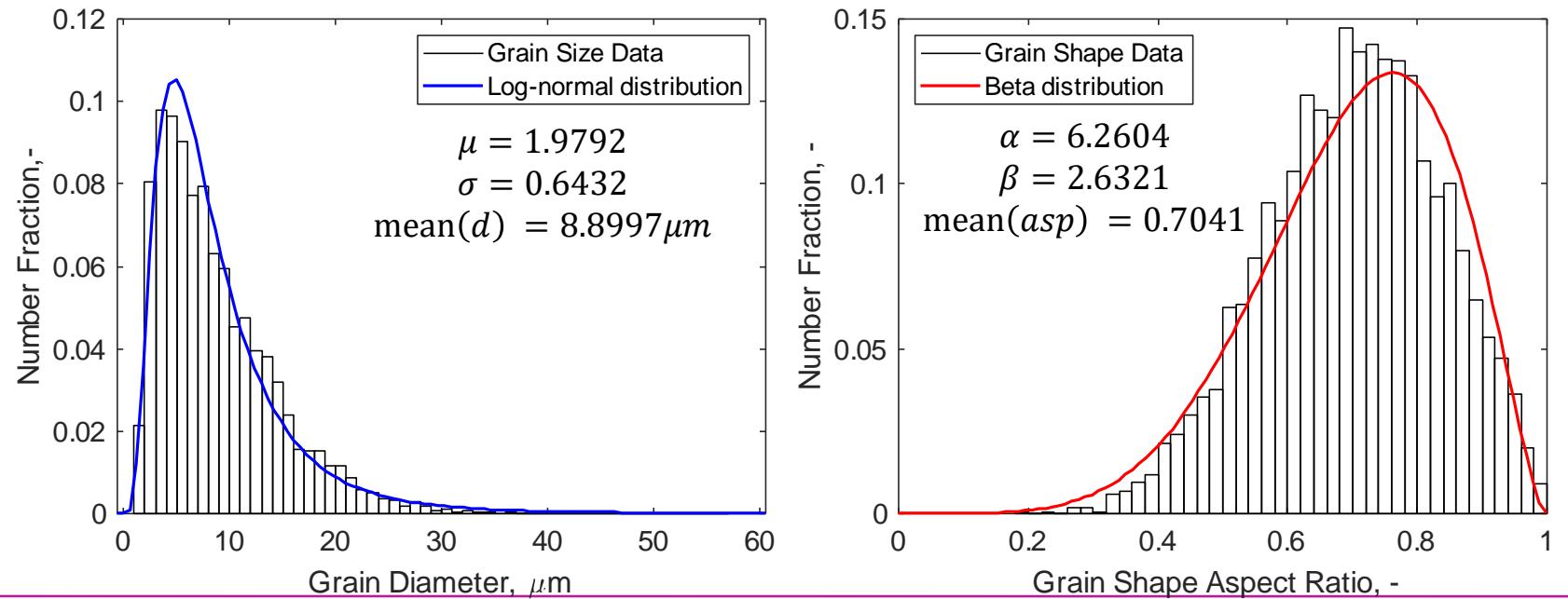
Grain size/shape distribution fitting

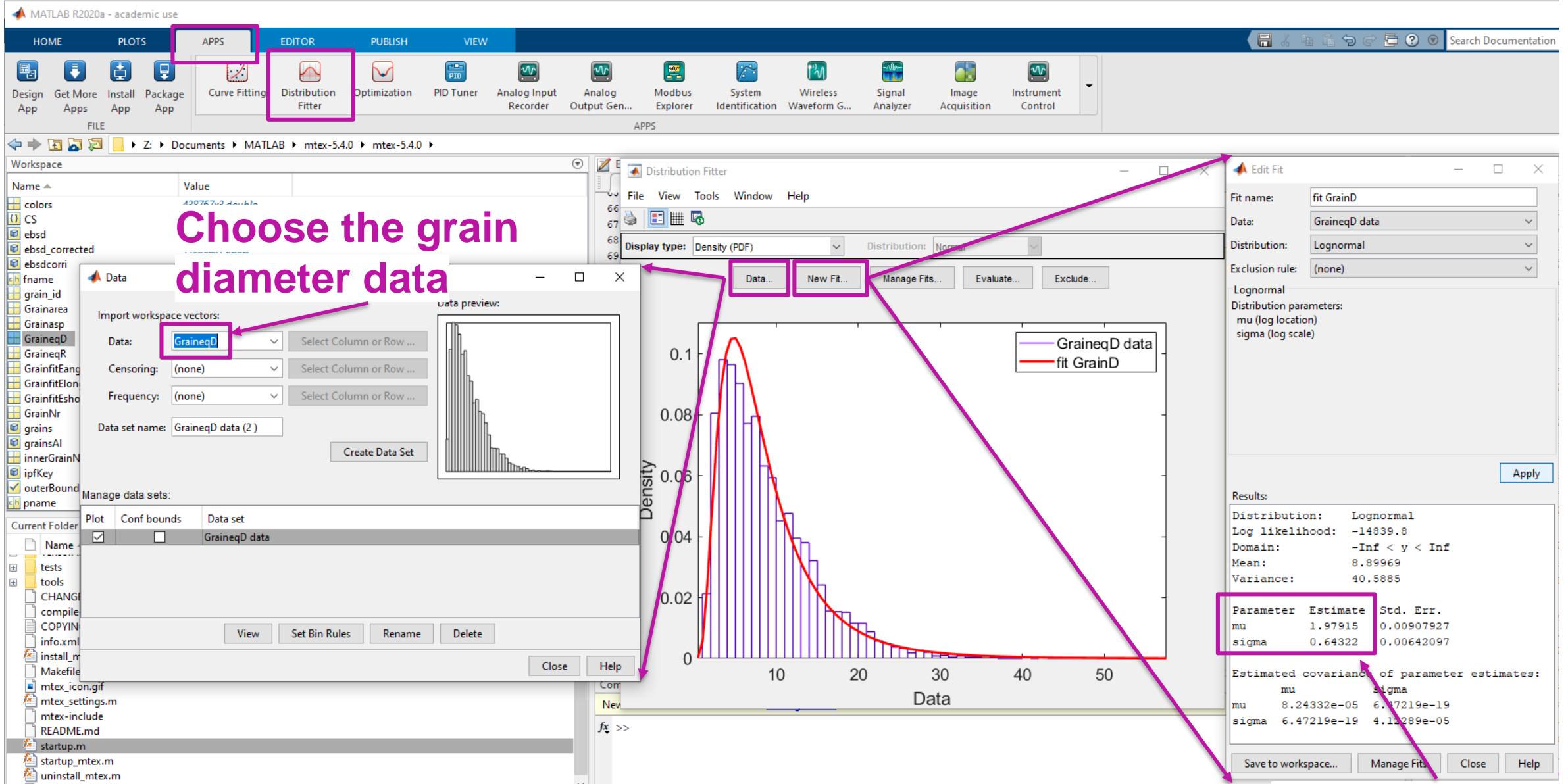
Grain size: log-normal distribution

Grain shape aspect ratio: Beta distribution

Tools:

- Matlab ‘Distribution Fitter’, ‘fitdist()’ function, etc.
- Origin
- Etc.





Here is only an example for grain size distribution fitting with the Matlab 'Distribution Fitter' application, please try to do the Beta distribution fitting with 'fitdist()' function by yourself for assignments.

Parameters for the fitted lognormal distribution.

Resources

Textbook:

- [1] Günter Gottstein, Physical Foundations of Materials Science, 2004.
- [2] William D. Callister, Jr. and David G. Rethwisch, Materials science and engineering: an introduction, 2011, 8th ed.
- [3] Olaf Engler and Valerie Randle, Introduction to texture analysis: macrotexture, microtexture, and orientation mapping, 2010, 2nd ed.

Software:

- Matlab (<https://se.mathworks.com/help/matlab/index.html>)
- MTEX toolbox 5.4.0 (<https://mtex-toolbox.github.io/>)
- ImageJ (<https://imagej.nih.gov/ij/>)

Questions?

- Assignment submission DL is **23:00 on 01.11.2020**.
- Contact: MyCourses ‘General discussion’ channel,
Wenqi Liu, wenqi.liu@aalto.fi