

MGranul Manual version 1.15

Introduction

MGranul is an implementation of correlation-based granulometry for multi-shaped objects, described in [Kim2012].

Its primary application is in characterization of SEM images or porous silicon.

MGranul is intended to be executed in Windows 7. However, it probably can be executed correctly in other Windows environments, such as Windows XP and Windows Vista.

If you want to reference this program in your paper, please cite:

[Kim2012] H. Y. Kim, R. H. Maruta, D. R. Huanca and W. J. Salcedo, “Correlation-Based Multi-Shape Granulometry with Application in Porous Silicon Nanomaterial Characterization,” *Journal of Porous Materials*, 2012 (to appear).

Installation

1) Uncompress mgranul115.zip preserving the subdirectory structure, in a directory, say, c:\mgranul.

2) Insert c:\mgranul\bin in path. This can be done:

MS-explorer → my computador → properties → advanced → environment variables

Where the path must be edited to append c:\mgranul\bin.

Main program

Execute:

```
c:\directory>mgranul
```

The program should answer:

```
< MGranul.exe: Programas para granulometria multi-formas v1.15>
```

```
Programas:
  Kernel - Gera imagem das mascaras a partir de kernel.cfg
  Correla - Maximos locais (.hol) das correlacoes com mascaras multiformas
  Mostra - Mostra maximos locais (.hol) sem filtrar
  Filtra  - Filtra maximos locais (.hol) e mostra
  OldCorrela - Maximos locais (.hol) das correlacoes - versao antiga
  Relat   - Le .ho2 e gera .rel
.....
Erro: Numero de argumentos invalido
```

Indicating that the program has 6 subprograms. “Erro” does not indicate an error in program, but that the parameters are missing.

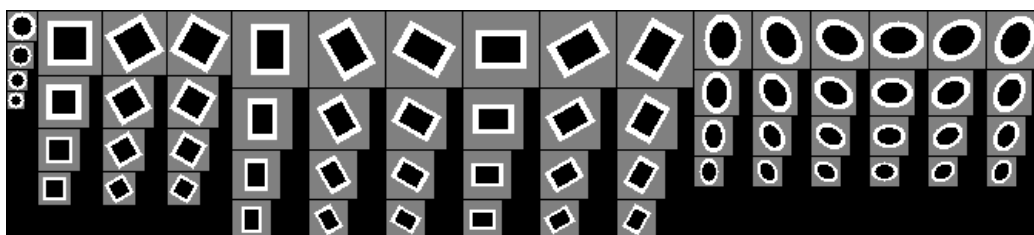
Subprogram Kernel

This subprogram generates the images with the kernels that will be used for searching for objects/grains. To test it, go to the directory c:\mgranul\exemplo_kernel and execute:

```
c:\mgranul\exemplo_kernel>mgranul kernel kernel.cfg kernel.png
or:
c:\mgranul\exemplo_kernel>roda
```

Subprogram kernel will read kernel.cfg and generate kernel.png.

```
//kernel.cfg
passoAng=30 // one template in each 30 degrees
escOitava=3 // 3 templates per octave
escala 0.5 a 1 piramide 2 //Generates the templates of
                          //all the following commands
                          //from scale 0.5 to 1.
                          //Searches for objects up to scale 2
                          //using pyramidal structure.
circulo 12 // Generates templates for circles.
           // Scale=1 corresponds to diameter=12
escala 0.5 a 1 piramide 1.4 //Generates the templates of
                          //all the following commands
                          //from scale 0.5 to 1.
                          //Searches for objects up to scale 1.4
                          //using pyramidal structure.
quadrado 20 // Generates square templates.
           // scale=1 corresponds to squares with width=height=20
           // Generates squares rotated from 0 to 90 degrees
escala 0.5 a 1 piramide 4 // Squares from scale 0.5 to 1
                          // Searches up to scale 4 using pyramid
retangulo 24 17 // Generates rectangles 24x17. From 0 to 180 degrees
elipse 24 17 // Generates ellipsis 24x17. From 0 to 180 degrees
```



kernel.png

At this moment, the implemented shapes are:

circle: no rotation

square: rotation from 0 to 90 degrees

rectangle: rotation from 0 to 180 degrees

ellipsis: rotation from 0 to 180 degrees.

The command:

```
escala 0.5 a 1 piramide 2
circulo 12
```

Will generate circles with diameters from 6 to 12 (corresponding to scales from 0.5 to 1). However, the searching will be executed in image at the original resolution and in image with half original resolution. This allows to find, in practice, circles from scale 0.5 to 2 (diameters 6 to 24).

Subprograma Correla

This program computes the peaks of cross correlation between the image to be processed and each kernel. To test it, go to the directory `c:\mgranul\exemplo_silicio` and execute:

```
c:\mgranul\exemplo_silicio>mgranul correla b54d.png quad.cfg b54d.ho1
```

The subprogram `correla` will read `b54d.png` and `quad.cfg` and generate `b54d.ho1` with the detected objects. The file `b54d.ho1` must be filtered to be useful.

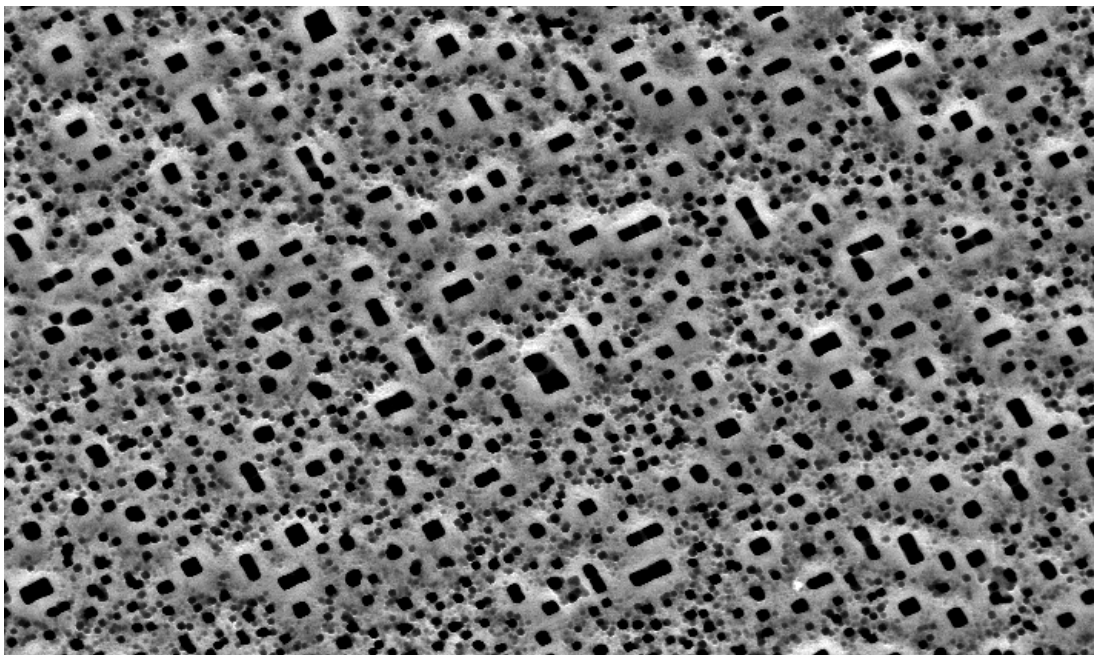


Image b54d.png

```
passoAng=10 // em graus
escOitava=5

escala 0.5 a 1 piramide 1
circulo 9

escala 0.5 a 1 piramide 1
quadrado 20 // lado*lado=400
retangulo 24 16.7 // lado1 ladoc
retangulo 28 14.3 // lado1 ladoc
retangulo 32 12.5
retangulo 36 11.1
retangulo 40 10
```

File quad.cfg

```

32680
0.646094 q 12 205 17.411 17.411 30
0.640164 q 74 623 11.487 11.487 30
0.633585 q 343 190 13.1951 13.1951 30
0.624020 q 204 114 15.1572 15.1572 30
0.612584 r 219 534 18.1886 12.6562 120
0.610443 r 352 589 21.22 10.8374 30
0.605101 q 25 288 13.1951 13.1951 30

```

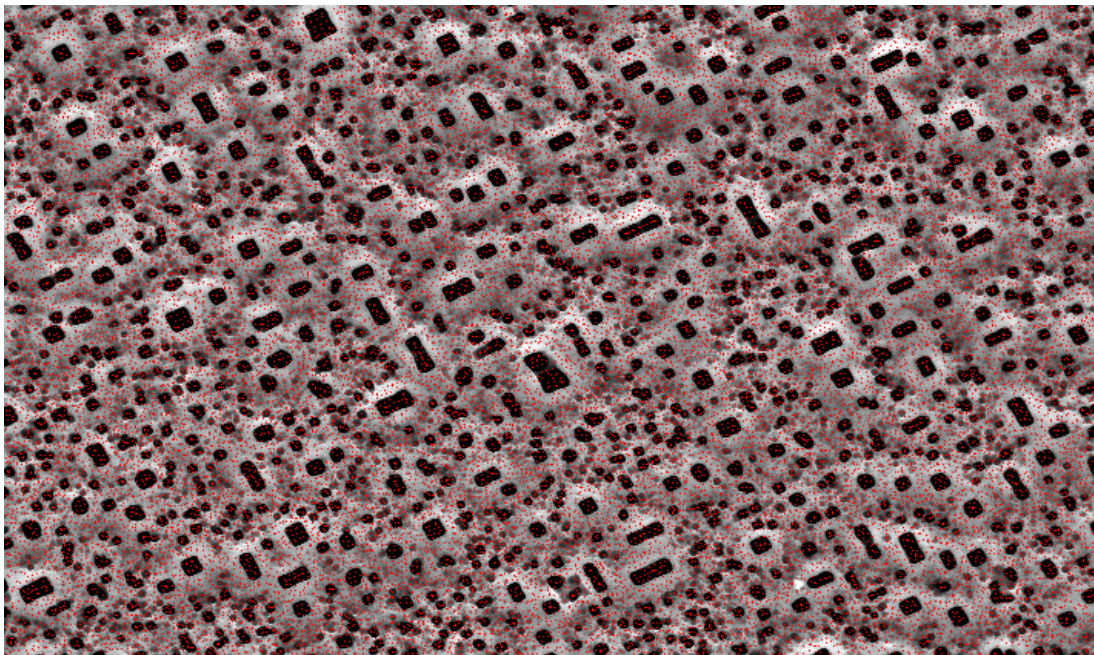
The first rows of b54d.ho1. The subprogram has found 32680 peaks. The second row indicates correlation=0.646094, shape=square (quadrado), y=12, x=205, side1=17.411 pixels, side2=17.411 pixels, angle=30 degrees.

Subprogram Mostra

In order to visualize if the file b54d.ho1 is correct, the following command can be executed:

```
c:\mgranul\exemplo_silicio>mgranul mostra B54D.png b54d.ho1 b54d_m.png s
```

This program will put a red point at each detected correlation peak. If replace “s” by “n” at the end of the command, the detected objects/grains will be delineated.



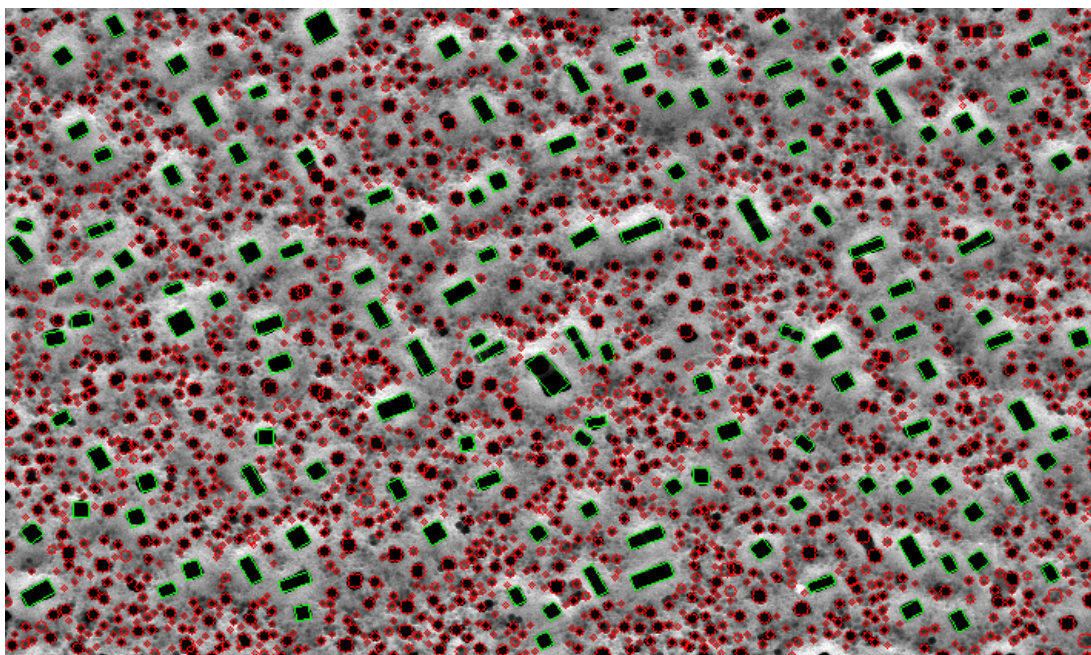
b54d_m.png

Subprogram Filtra

Clearly, the subprogram Correla has detected an excessive number of peaks in b54d.ho1. To filter it, the subprogram filtra must be used:

```
c:\mgranul\exemplo_silicio>mgranul   filtra   b54d.png   b54d.ho1   b54d_f.png  
"circulo 0.1 0.5 quadrado 0.45 0.05 retangulo 0.45 0.05"
```

The last argument enclosed by “” are τ (minimum correlation threshold) and γ (maximum intersection threshold) for each shape. For example, circulo 0.1 0.5 indicates that circle shapes have parameters $\tau=0.1$ and $\gamma=0.5$. These parameters must be adjusted interactively by the user.



The subprogram filtra can generate output image or output report. To generate the report, execute:

```
c:\mgranul\exemplo_silicio>mgranul   filtra   b54d.png   b54d.ho1   b54d_f.ho2  
"circulo 0.1 0.5 quadrado 0.45 0.05 retangulo 0.45 0.05"
```

The only alteration is the extension of the output file to .ho2. This extension indicates that the report must be generated.

```
2331
0.646094 q 12 205 17.411 17.411 30
0.640164 q 74 623 11.487 11.487 30
0.633585 q 343 190 13.1951 13.1951 30
0.624020 q 204 114 15.1572 15.1572 30
0.612584 r 219 534 18.1886 12.6562 120
0.610443 r 352 589 21.22 10.8374 30
0.605101 q 25 288 13.1951 13.1951 30
0.589196 r 137 486 31.3398 9.66311 30
0.588830 q 38 463 10 10 30
0.588726 r 379 21 21.22 10.8374 120
0.586394 c 329 193 5.93779 5.93779 0
```

The first rows of b54d_f.ho2, with 2331 detected objects. The second row indicates correlation=0.646094, shape=square (quadrada), y=12, x=205, side1=side2=17.411 pixels, angle=30 degrees.

The “pattern spectrum” can be plotted using this report.

Re-compilation of Granul

To compile mgranul.cpp, you have to install Cekeikon library (www.lps.usp.br/~hae/software). Then, you can compile it using:

```
c:\mgranul\src>ccek mgranul
```