**The procedure to calculate image’s Pn function, Omega Matrix**

**and Differential Omega Matrix (DOM).**

INTRODUCTION

In this document, our team will illustrate how to use the programmed code to calculate image’s Pn functions, Omega Matrix and Differential Omega Matrix (DOM).

For the concise and clear purpose, we will describe the procedures step by step. Before we start to run the programming, I want to introduce the software used through the whole calculation.

SOFTWARE

The first software I use is MATLAB. I use MATLAB as an image processor to let the RGB image to be transferred a digital matrix which only contain 1 and 0. 1 means the interested phase/particles and 0 means void in the original mage. The setting and its special features can be found in the step 3.

The second software I use is Visual Studio. For efficient purpose, we suggest that one can use super computer or cluster to calculate the results of image(s).

MAIN FEATURE

The programmed codes revised and created by our team can offer several multi-function ability to cope with deferent condition to describe, predict and design the microstructure of interesting object. You can check the related papers in journal to fine how these codes work. In order to let the codes equip the mentioned abilities, our codes need 2 significant features.

1. Any image of evolution process can be selected as the start point/reference. One doesn’t need to rename the all files according to different cases. It will save user plenty of time.
2. One can easily select special area of image to analyze its microstructure. Under this situation, anyone can research the localized configuration easily and doesn’t need to modify the image to desired scope or size for various research targets.

EXAMPLE

In order to demonstrate the whole procedure in detail, we put the image files image.0018.png, image.0019.png, image.0020.png, image.0021.png and image.0022.png which present the images of object’s evolution from its 18th step to 22th step to illustrate its ability that “one can select any image as the start/reference point” in MATLAB code. By selecting difference referenced image according to the design requirement, our codes have more accommodated ability to offer more simple way to let it implemented.

One can check the underlined part in step 3 to change the value of Reference to select the referenced image. The default value of it is 18 in the digitization.m because we want to set the 18th image as the start point. If you don’t change this value and run the code. 1Mconfig.txt obtained from running digitization.m file means that we select 18th image as our first/referenced image and the content of 1Mconfig.txt is its digital matrix.

One can check the underlined part in step 3 and step 4 to select the specific area to be analyzed. Take image.0018.png file of which size is 472 by 284 pixels for example, the interested square scope of this image is from (42,136) to (242,336) because of the limitation of the given image(shown as Figure 1 ). By using these codes, this 200 by 200 pixels area can easily be select by end user to save lots of time.

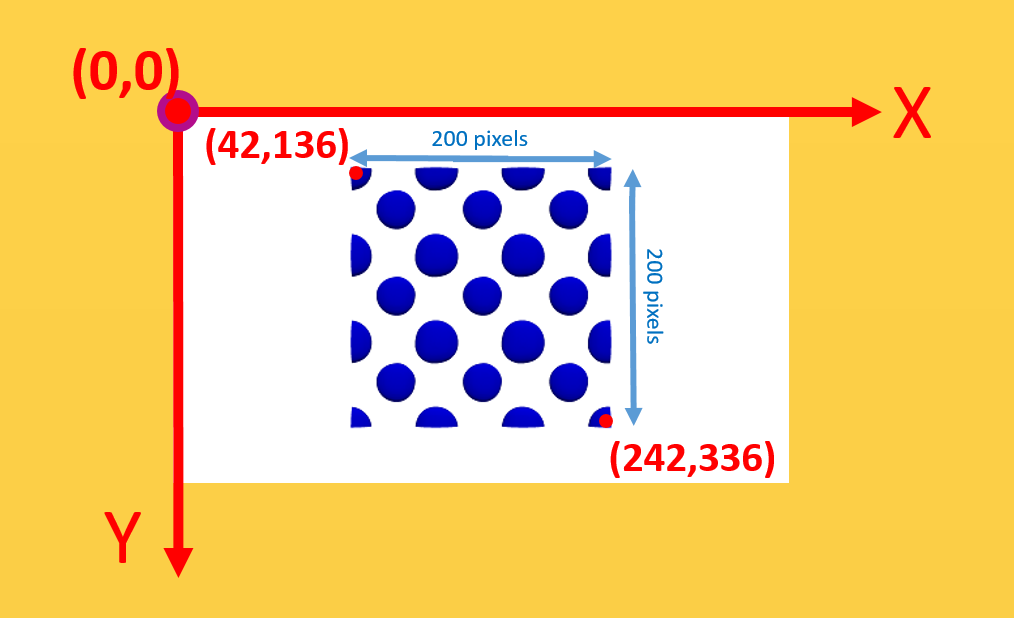


Figure 1 .The scheme of selecting the specific area for image.0018.png.

The following steps are the procedures to obtain the results of image’s Pn function, Omega Matrix and DO).

THE PROCEDURE TO CALCULATE Pn, OMEGA MATRIX AND DOM

1. Selecting the square image

Since our team will use symmetric models such as regular triangle, square, regular, regular hexagon and regular Octagon to obtain image’s Pn functions and Omega Matrix, I only segment its hidden microstructure’s features from its square image based on the consideration of symmetric basis.

1. Numbering the image(s).

The default format of image file used for calculation is “image.XXXX.png”. For evolution analysis, the image is numbered from “image.0001.png”. According to the code’s setting, you can input 9,999 images into code.

**Note:** If you have different input file format, you can modify the content of code named as digitalization.m in line 13.

1. Opening the file named as digitization.m located in fodder digitization of by MATLAB

In this file, you enter the parameters according to your research. The following descriptions are important features that you need to know before you run the file.

＃PicNum means that you can choose many images you want to use by enter the number in line 4.

＃Reference means you want to set which image as the reference by enter the

number in line 6.

＃For obtaining the accurate result reason, we suggest you segment the different phase of image before you import the image into MATLAB. In the digitalization.m file, you also can adjust the value of threshold to segment phase for simple cases.

Taking the value of threshold=230 for example, the final segmentation result is reasonable compared to the case of which value of threshold=20 when you enter 230 and 20 separately and run the digitalization.m file. The result of adjust the value of threshold is shown as Figure 1. One can adjust this value to get the most decent results.

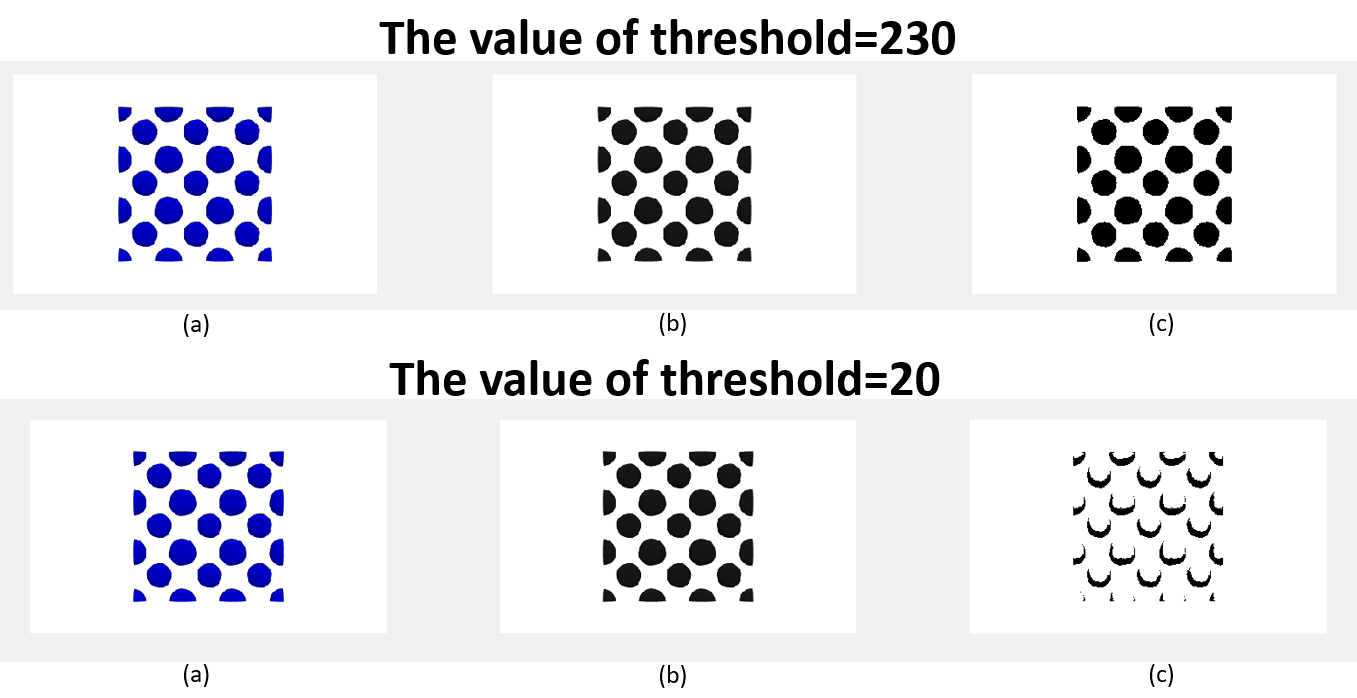


Figure 1 .The results of different value of threshold. (a) The original input image. (b) The grayscale image of original input RGB one.(c) The final result of image segmentation.

＃You can select any part of the input image by changing the values located in line 65. The only thing you need to double check is that the selected scope must be square. The size of example image named image.0018.png is 472 by 284 pixels. I select the square scope of it by setting the valued located in line 65 to obtain a 200 by 200 pixels image to proceed my research.

After you finish the parameters setting and run the code. The files named █Mconfig.txt where the █ represents the digital number from 1Mconfig.txt will be generated in the same fodder. These Mconfig.txt is the digitalized results of images.

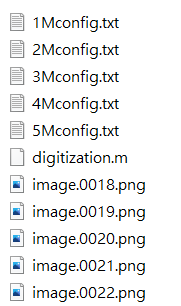


Figure 2. The final results when you run the digitalization.m file.

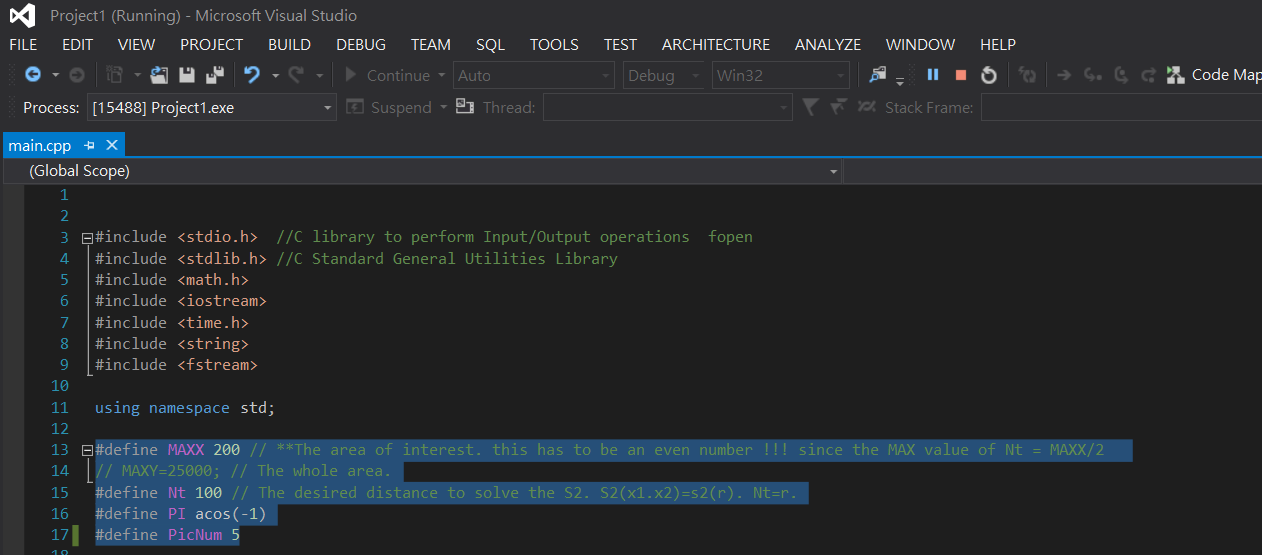
1. Obtaining the images’ Pn function.
   1. Coping and pasting all the █Mconfig.txt files from the 01-digitization to the 02-Pn fodder, 03-OmegaMatrix and 04-DiffOmegaMatrix to proceed the calculation processes of Pn functions, Omega Matrix and Differential Omega Matrix.
   2. Click the file named Project1.vcxproj located in the fodder of 02-Pn then adjust the parameters located from line 13 to 17 under the main.cpp file.

＃The MAXX is the size of side of the input image in pixel.

＃Nt is the desired distance to solve the Pn functions and its value is always equal the value of MAXX.

＃PicNum is the number of input image.

＃If you want to only analyze specific area of image, you need to enter the coordinate of the most left-upper point in line 928 and line 929. If you don’t need it, just delete these 2 lines.

Figure 3.The parameters needed to modify are located from line 13 to 17.

* 1. After finishing the setting of the parameters, you can run this code and you will get image’s results of Pn functions (shown as Figure 4).

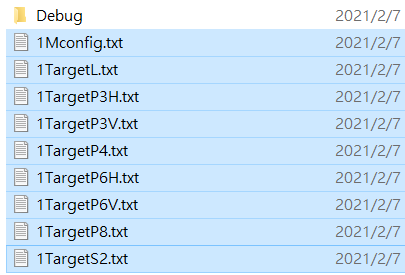


Figure 4. The result of Pn functions of first image. 1TargetL means the Lineal-path function of image 1. 1TargetS2 means the P2 function of image 1. 1TargetP3H means the P3 horizontal function of image 1. 1TargetP3V means the P3vertical function of image 1. 1TargetP4 means the P4 horizontal function of image 1. 1TargetP6H means the P6 horizontal function of image 1. 1TargetP6V means the P6vertical function of image 1. 1TargetP8 means the P8 function of image 1.

1. Obtaining the images’ Omega Matrix.
   1. Click the file named OmegaFunction.vcxproj located in the fodder of 03-

OmegaMatrix then adjust the parameters located from line 11 to 15 under

the MainO2.cpp file.

＃The MAXX is the size of side of the input image in pixel.

＃Nt is the desired distance to solve the Pn functions and its value is always equal the value of MAXX.

＃PicNum is the number of input image.

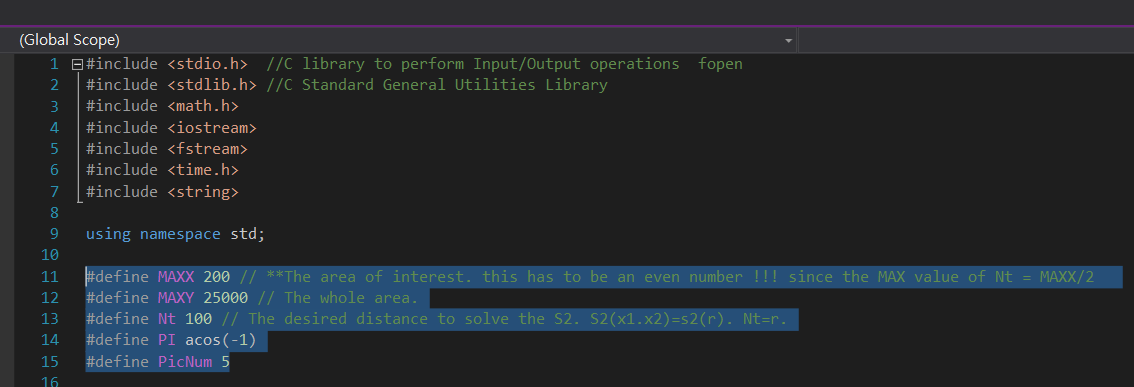


Figure 5.The parameters needed to modify are located from line 11 to 15.

* 1. After finishing the setting of the parameters, you can run this code and you will get image’s results of Omega Matrix (shown as Figure 6).



Figure 6. The result of Omega Matrix of all input images.

1. Obtaining the images’ Differential Omega Matrix.
   1. Click the file named OmegaFunction.vcxproj located in the fodder of 04-

DiffOmegaMatrix then adjust the parameters located from line 11 to 15

under the mainDiff.cpp file.

＃The MAXX is the size of side of the input image in pixel.

＃Nt is the desired distance to solve the Pn functions and its value is always equal the value of MAXX.

＃PicNum is the number of input image.

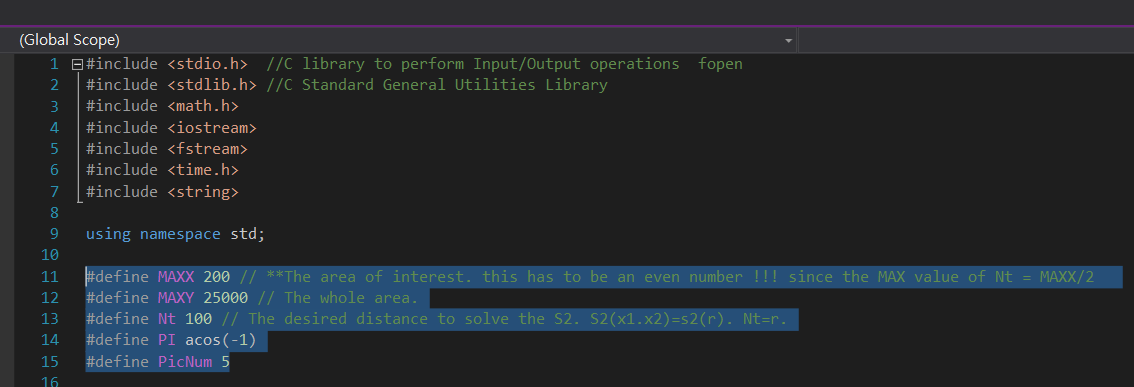


Figure 7.The parameters needed to modify are located from line 11 to 15.

* 1. After finishing the setting of the parameters, you can run this code and you will get image’s results of Omega Matrix (shown as Figure 8).

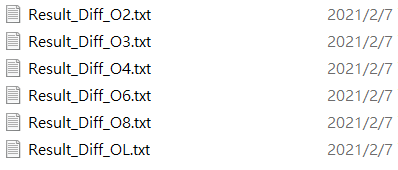


Figure 8. The result of Differential Omega Matrix of all input images.

Above are the steps to calculate image’s Pn function, Omega Matrix and DOM. One need to check the parameters such as MAXX, Nt and PicNuM in all cpp files are the same, otherwise the software will feedback wrong message to you because the matrix maybe overflows its original set size.