

# Designing and Programming an Efficient Software for Sizing and Counting Various Particles using Image Processing Technique

*Seyed Vahid Seyedin*

Dep. of Computer Engineering, Islamic Azad University, South Tehran Branch  
No. 223, Headquarter of Islamic Azad University, South Tehran Branch, ZIP area 11, Azarshahr Street,  
North Iranshahr Street, Karimkhan-e-Zand Avenue, Tehran, Iran  
Phone: + (98-21) 88830826-30

*Seyed Hadi Seyedin*

Dep. of Chemical Engineering, Islamic Azad University, Science and Research Branch  
Science and Research Branch, Daneshgah Blvd, Simon Bulivar Blvd, Tehran, Iran  
Phone: 44865179-82  
shadi.seyedin@iran.ir

*Atiyeh Sadat Seyedin*

Dep. of Biotechnology Engineering, University of Tehran, Faculty of New Sciences and  
Technologies  
Tehran Province, Tehran, Enghelab Square, Iran  
Phone: +98 21 61111

## Abstract

Main purpose of the present study is to develop an appropriate computer program to size and count particles in digital images focusing on eliminating undesirable objects from the image. In this study, the user friendly Graphical User interface (GUI) tool is developed using MATLAB to measure the size of every particle of the digital image. The step by step process optimizes synthesis process of micro particles. It also facilitates generation of the required statistical analysis for medical science, material science and other applications. The described image processing system reduces total measuring time, eliminates subjective observer error in sizing and determines fine particles, considerably. This method can be used as a simple pattern recognition technique to size and count a relatively large number of particles in a short time. It is applicable to filter out images of undesirable objects such as redundant spots or noises in the image and clearly, by taking high resolution photos, the result of the processed images would be more reliable.

**Keywords:** Counting Particles; Image Processing; Particle Size Distribution; Image Segmentation; Cell Morphology; Medical Imaging; MATLAB.

## 1. Introduction

Particle size distribution of granular or powder materials is usually evaluated for research and experimental works. Most of the systems used for particle measurement such as light microscopes are too expensive and they are not available everywhere. One of the approaches proposed to overcome this problem is to use image processing as an in-situ test method because this method only requires a camera and a computer and it is a fast and cheap method for measuring size of particles.

In recent years, production and application of particles have increased extensively due to their unique physical and chemical properties (Rajeshwari et al., 2013). In order to ensure good use of resources from environmental and economic aspects, the quality of materials should be characterized. Particle size distribution of materials is one of the tests used widely in new sciences such as nanotechnology, biotechnology, material science, metallurgy and geotechnical engineering to evaluate quality of materials. Size of particles is measured using complementary techniques like X-ray diffraction (XRD), scanning electron microscopy (SEM) and light microscope providing the average information from a large number of particles instead of a single crystal. To our knowledge, using these techniques for a single crystal results in difficult interpretation (Reetz et al., 2000;

Coster & Chermant, 2001). Image analysis techniques are extensively used in medical and electronic industries (Coster & Chermant, 2001). Also, this method gives rise to larger diameter and cross-sectional area of nanoparticles than those obtained from one-by-one inspection in SEM and TEM images (Phromsuwan et al., 2013). Furthermore, using this approach, images are analyzed with little human intervention and the images are more accurate and more objective than those obtained using a global threshold. As a natural extension of the concept, the results can be greatly improved by applying an adaptive threshold to selected regions of the images interactively (Gontard et al., 2011).

In addition to shape characteristics, size characteristics of aggregates including particle size distribution with image analysis have also been discussed in the past (Fernlund, 1998; Kumara et al., 2011). The image processing method directly calculates inter-particle distance of suspension and aerodynamic particle size distributions from pharmaceutical inhalers. This direct measurement method is more reliable than previous theoretical models (Fishler et al., 2018; Maiti et al., 2017). Digital image processing generates size distribution of particles which is fast and accurate and it is often expressed in terms of area function or number of particles. In general, 2-D images are used for image analysis. However, volume of particles cannot be measured directly in 2-D images. The results showed that 2-D images give good results when particles are placed on a stable location. However, it is a time-consuming process due to the fact that location of particles should be changed manually twice to obtain images for measuring dimensions of the three axes. As mentioned in the literature, volume of particles is determined using 2-D images with some assumptions or using a time-consuming process, for example, measuring mass of individual particles (Banta et al., 2003).

The real-time image technique of sensing and processing is one of the most promising methods for sizing and counting small particles. By applying this technique, one can store real-time images in a computer through a digital camera so that the stored information can be analyzed automatically (Ow, 1981; Ahler & Alexander, 1985). The main advantage of this technique is that measurement and analysis can be performed in a very short time. The other advantage of this technique is elimination of a subjective human error which is almost unavoidable in manual sizing and counting (Weiss et al., 1984). In spite of these advantages, there are some basic problems related to system resolution and light diffraction (<https://www.mathworks.com/discovery/digital-image-processing.html>).

In the present research, images of the particles have been captured. Then, diameters and numbers of particles have been measured and counted using image processing and pattern recognition techniques. Since MATLAB is an advanced procedural language and user friendly tool for data analysis and mathematical computing function, this software has been used as a powerful programming tool for this purpose. The image processing toolbox provides a wide collection of advanced image processing functions and interactive tools for enhancing and analyzing digital images. The toolbox functions implemented in the open MATLAB language have been used to build up the customized algorithms (Hahn & Valentine, 2016). It is known that image processing techniques can be easily used to implement complex image processes, mainly for fast prototyping (Fernlund, 2005), so a simplified algorithm has been proposed to determine size of particles from digital images. To provide statistical information from the image, the software provided by means of MATLAB GUI and programming knowledge have been utilized. In this study, an image processing method is developed for calculating particle sizes directly, which is tremendously practical for particle sizing and counting digital images in various science researches.

## **2. Methodology**

### **2.1. Particles**

The objects including odd-shaped foreign materials, in contact or overlapping particles, parts of single particles cut off by the image frame boundaries and noises in image background are shown in Figure 1.

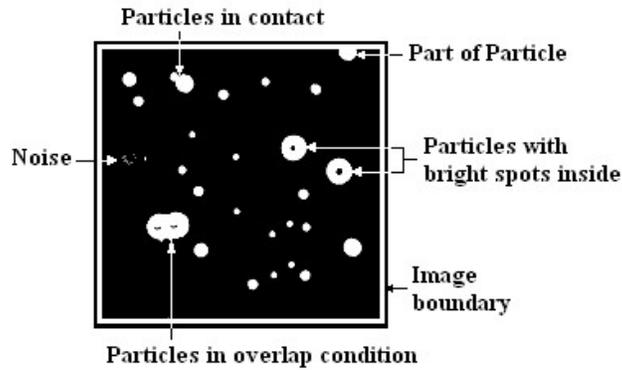


Figure 1. The schematic image of particles containing undesirable objects in background

Particle sizes varying from 500  $\mu\text{m}$  to 1 mm are used in laboratory test and image analysis. The particles have various shapes and the mean diameter of particles in 2D\* is computed by image processing method. In this work, the particles are put on a black background. Before processing by computer, the original images are converted to BW (Black and White).

## 2.2. Photographic System

The system has two parts. A light source, a digital camera and a computer program for image sensing and processing operations to determine particle size distribution. The system has a digital camera also (Canon model EOS – 60D) and an appropriate lens (Type EFS 18-135, f3.5 - 5.6) with automatic focus used to obtain high resolution images.

## 2.3. Image Capturing Process

Particles are arranged manually such that particles stand on a stable position. Similar arrangement for aggregates of 35mm diameter were used in (Kumara et al., 2012; Pratt, 1978). In this arrangement, thickness (i.e., short axis) of particles cannot be measured directly, since 2-D images measure only the long and intermediate axes. In this research, particles are arranged without touching or overlapping each other to reduce application of unnecessary image processing techniques and increase accuracy. Particles are placed on a black sheet and a suitable light source is used to obtain photographs of particles with better contrast during photography process. The main light was back light.

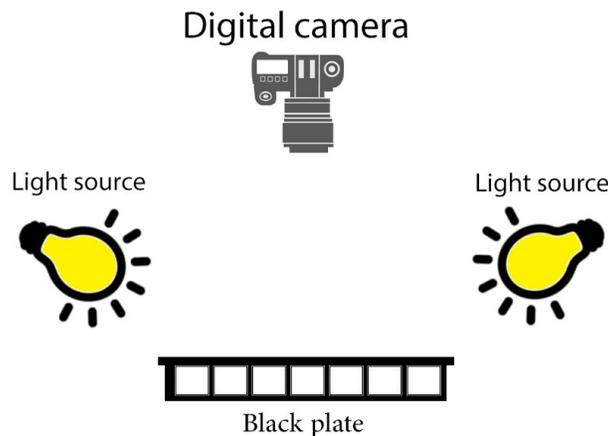


Figure 2. Image capturing system and various positions of the light source

It can be seen in Figure 2 that the top light is applied to strengthen back light effect with an attracted light sheet (black sheet). A scale (ruler) is also used during image analysis process. This

\*. Two Dimentional.

system includes a digital camera for photography and a computer program to control the image processing performing the pattern recognition algorithm.

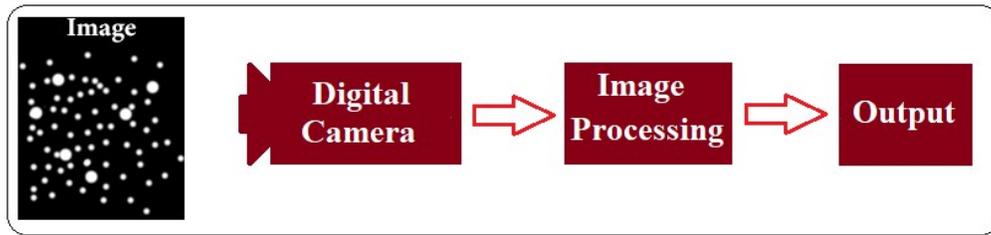


Figure 3. Block diagram of image sensing and processing system

In the present study, the digital images have a resolution of 1494 x 2656 pixels (picture elements) with 256 brightness levels ranging from zero (black) to 255 (white).

### 3. Image Processing

#### 3.1. Image Processing System

As mentioned before, a GUI (Graphical User Interface) software is programmed by MATLAB R2014b. Image analysis is done in MATLAB. MATLAB can read many image formats including TIFF, GIF, JPEG, BMP, DICOM, FITS and JPEG which are used in this research. MATLAB can calculate area and pixel value statistics of user defined selections. It supports standard image processing functions such as contrast manipulation, sharpening, smoothing, edge detection and median filtering.

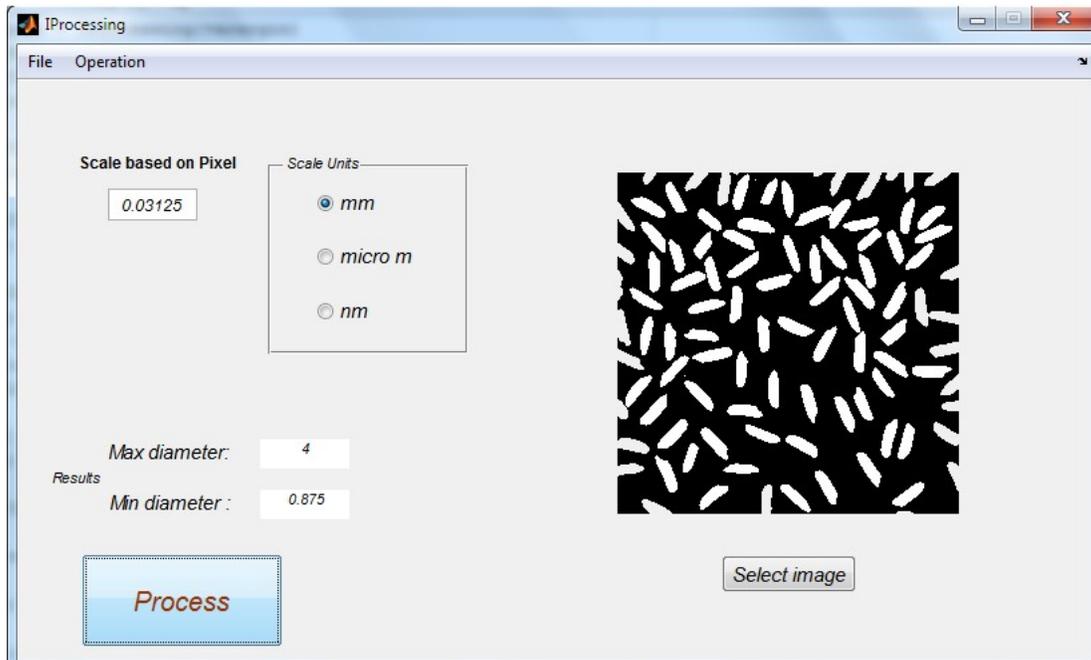


Figure 4. The GUI software that allows the user to measure selected particles

In image processing, the pixel region tool is provided by MATLAB to set on each pixel region over image and it is displayed in the image tool. The major group of displayed pixels can be seen in extreme close-up using the pixel region tool in a new window. Current position of the pixel region in the specified image can be approximated by checking the pixel information. Thus, pixel coordinates (x and y) in the board image coordinate system is obtained (Figure5).

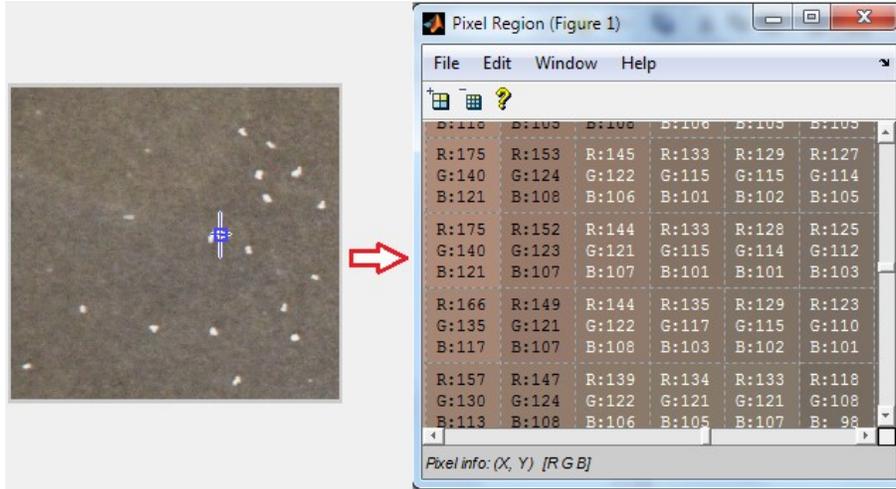


Figure 5. The GUI program that allows the user to investigate pixel information of selected image

To illustrate the significance of this problem, an example is shown in Figure 6. Two solid circles with equal diameters are considered. In Figure 6, brightness level histogram of two circles is shown. The number of pixels corresponding to diameter of the object depends on the selected threshold level. However, if the threshold level is lowered to a brightness level of 100, the number of pixels representing diameter of the left circle will be 50 whereas diameter of the right circle will be 51. If the threshold level is raised to a brightness level of 175, the number of pixels representing diameter of the right circle becomes larger than the left circle. The proper threshold level can be determined from distribution of the brightness levels in the digital image frame (Gonzalez et al., 2004). For intensity images,  $n$  bins of the histogram are half-open intervals with width of  $A/(n-1)$ . In particular, the  $P^{\text{th}}$  bin is the half-open interval.

$$A(P-1.5)/(n-1) \leq x < A(P-0.5)/(n-1) \quad (1)$$

The input image can be logical, uint8, uint16 or double (Kim & Lee, 1990). Scale factor,  $A$ , depends on the image class.  $A$  is 1 if the intensity image is double and 255 or the intensity image is uint8 or uint16 class support. Figure6 shows histogram of pixel values at the top of each color bar of the color map.

Before image processing, the original image must be converted to BMP format because the BMP file format is capable of storing 2D digital images with arbitrary width, height and resolution. In MATLAB, original images can be converted into binary images. Usually, it is assumed that binary images have white objects and black background. Binary images are very important to make some process by programming commands, such as erode, dilate, open, close, fill holes, water shed and etc. Erode removes pixels from the edge of black objects. Dilate adds pixels to the edges of black objects. Open performs an erosion operation followed by dilation. Close performs a dilation operation followed by erosion. Fill holes fills holes (empty space) in objects. Due to some light effects, there might be white spots in black objects. These white spots should be eliminated using fill holes. Water shed separates or cuts touching particles. In this research, particles are arranged without touching or overlapping each other.

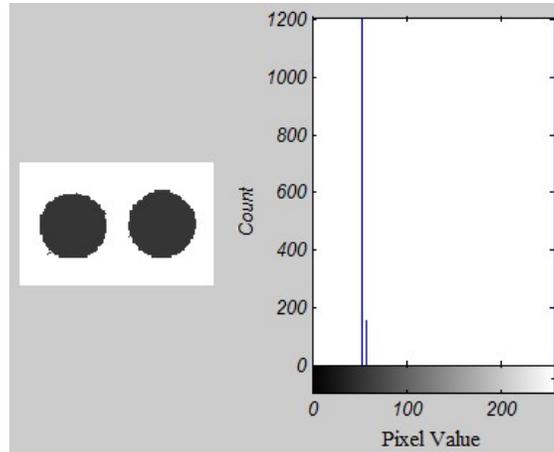


Figure 6. The intensity histogram of two solid circles with equal diameters

Figure 7 shows the image analysis process. First, pixel values are converted into mm using a scale factor (ruler).

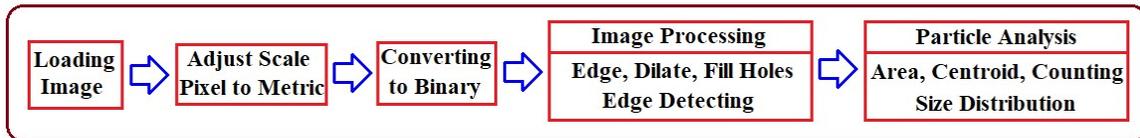


Figure 7. Image analysis process

36 various particles are placed on a black color sheet. The shadow effect and quality of digital images for comparing the results are investigated. These digital images with different brightness levels are converted to binary images of BW (black and white) based on the threshold level. To our knowledge, one should take care in choosing the threshold level so that it does not distort the information (particle diameters in the present case) of the original images; several methods have been proposed in the past (Castleman, 1979). Calibration of scale factor is done with the scale placed on the images as shown in Figure 8. Then, images are converted into binary images after unnecessary parts of the images are cut without disturbing particles in the images to eliminate any bad light affecting edges of the images if any bad light exists. Figure 8 shows original image of jpg type (4096 x 3072 pixels) and binary images of particles, respectively.

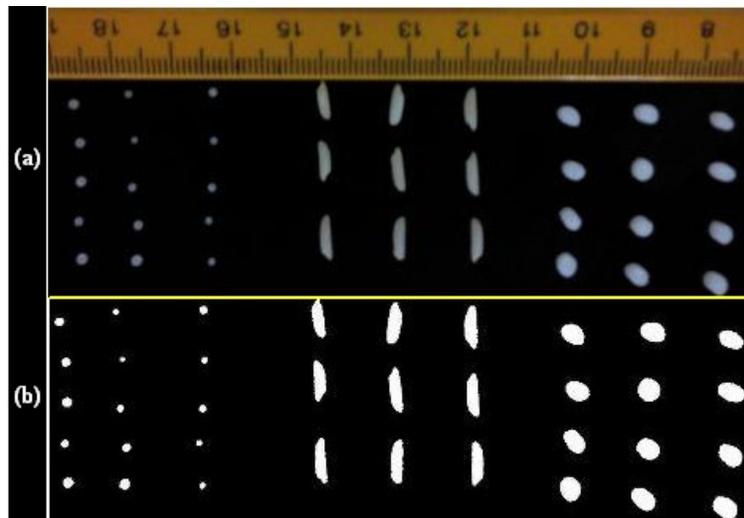


Figure 8. (a) Original image and (b) binary image of sample particles on black color sheet

### 3.2. Pattern Recognition Algorithm

In order to identify particles from the binary images, connectivity of pixels with the same brightness level should be considered. In the present study, pixels indicating the objects (particles) are considered to have 8-connectivity whereas the pixels indicating background are considered to have 4-connectivity (see Figure 9). In the 4-connectivity, only four adjacent pixels in the horizontal and vertical directions are considered. On the other hand, in 8-connectivity, four adjacent pixels in the diagonal directions are taken into account in addition to the pixels considered in 4-connectivity. The line-by-line method is used as a basic scheme of the pattern recognition algorithm to distinguish the objects from the background (Park, 2018). In the line-by-line method, several adjacent rows are simultaneously taken into account for analysis and this procedure is continued sequentially from the top to the bottom line of the image. In Fig 9, two connectivity models are presented for calculating the object size. The particles (objects) are white and image background has black colors (BW image). Thus the 0 and 1 matrix is used for image processing and SE is the structure element (objects in the image) in MATLAB.

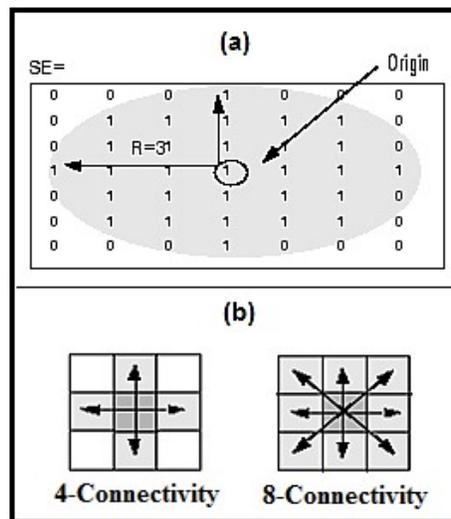


Figure 9. (a) The 0 and 1 matrix of BW image, (b) Two types of pixels connectivity

The cases of abnormal connectivity are observed when the image contains undesirable objects such as particles in contact or odd-shaped foreign materials. Therefore, if this abnormal connectivity is properly handled, undesirable objects during pattern recognition process can be automatically filtered out. During isolation and image modification process, in which the patterns are recognized, the digitized binary image is modified. Through image modification process, brightness level of the blank space (hole in an object) inside the object boundary is converted into the brightness level of the runs (dark) and all voids are eliminated. In addition, the overlapping particles are converted into a single particle. Then the image of the two particles in contact is converted into two distinct parts (see Figure 10).

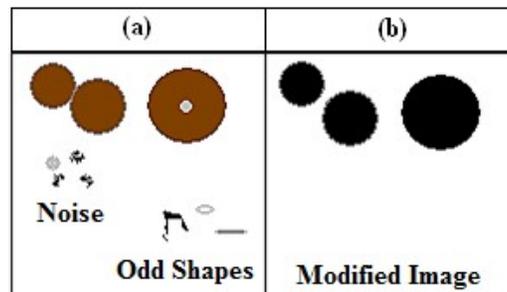


Figure 10. Process of filtering the image. (a) Original image; (b) isolation and image modification

Based on quality of the images judged by the operator, number of steps for modification such as dilate, fill holes and erode processes are decided. It is important that number of dilate and erode process steps is always equal to ensure that pixels of real particles are not increased or removed.

### 3.3. Labeling and Measuring Objects in a Binary Image

A connected component in a binary image is a set of pixels that form a connected group. For example, the binary image below has three connected components. Connected component labeling is the process of identifying the connected components in an image and assigning a unique label to each one that is called a label matrix (see Figure 11).

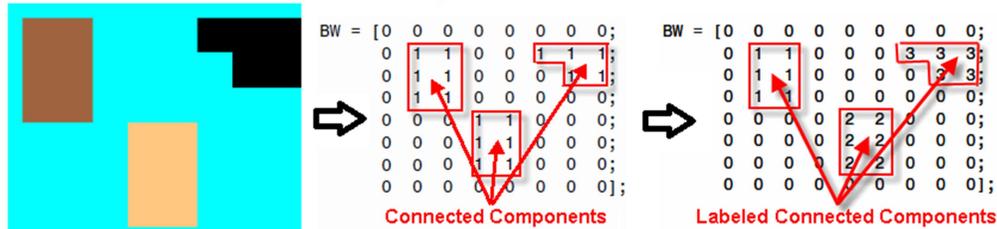


Figure 11. The label matrix of some objects in the background

The overall algorithm for pattern recognition of image processing is shown as the flow chart in Figure 12.

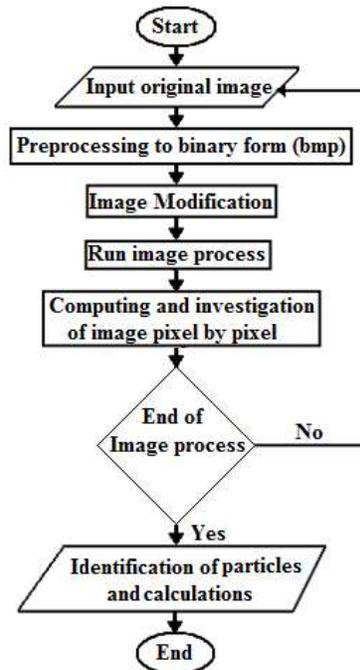


Figure 12. Pattern recognition flowchart

### 4. Test of the Recognition Program

To check performance of the present measurement system, two sample cases are tested. In the first part of the test, the reference picture in Figure 13 which contains some objects of known size, is prepared as a specimen.

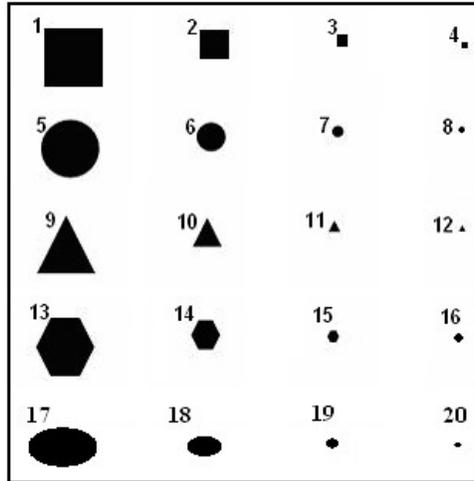


Figure 13. Reference picture of 20 objects for checking the program

Results of manual counting and sizing are shown and compared in Table 1.

Table 1. The results of reference image processing and comparing to actual value

Object No.	Real value		Calculated by software		Difference (%)
	Mean diameter (mm)	Area (mm <sup>2</sup> )	Area (Pixel)	Area (mm <sup>2</sup> )	
1	10	100	900	99.8	0.2
2	5	25	225	24.95	0.2
3	2	4	36	3.99	0.25
4	1	1	11	1.1	9
5	10	78.5	710	78.1	0.5
6	5	19.625	175	19.32	1.554
7	2	3.14	30	3.32	5.42
8	1	0.785	9	0.81	3.1
9	5	50	447	49.56	0.88
10	2.5	12.5	119	13.1	4.58
11	1	2	19	2.1	4.76
12	0.5	0.5	6	0.66	24.242
13	10	75	669	74.18	1.09
14	5	18.75	174	19.29	2.8
15	2	3	26	2.8	6.67
16	1	0.75	9	0.81	7.4
17	10	150.72	1350	149.7	0.67
18	5	37.68	337	37.425	0.66
19	2	6.0288	54	5.985	0.72
20	1	1.5072	16	1.8	9.04

Also, as shown in Table.1, error range is observed for each shape group. This may be due to the discretization error originating from the process of converting the original digital image into the binary digital image; however, there might be a certain error involved in making objects when preparing the reference picture. The actual area is calculated using the mathematical formula for various geometrical shapes.

## 5. Results and Discussion

The measure tool is projected to aid measuring the size of particles in an image. To measure the image, little illustration is required to calibrate the pixel using a scale before processing.

### 5.1. Image Processing Toolbox

The tool is planned for high quality measurements and it uses image processing toolbox of MATLAB. These tools allow the user to select, zoom, readjust and confirm the selection. It also allows placing the measurement tool which can be customized using its control points. In this GUI measure tool, a set of plotting options can be established by setting it. The intensity is calculated by plotting the pixel values on discretized points of the measured line and its averages. For a particle, the average intensity is calculated for every pixel within the particle.

### 5.2. Sizing and Counting Particles

Figure 14 is the number histogram of sizes of the 36 particles obtained from the reference picture shown in Figure 8 with the unit pixel length being set to 0.03125 mm. The unit pixel length is based on the value in scale on the original image.

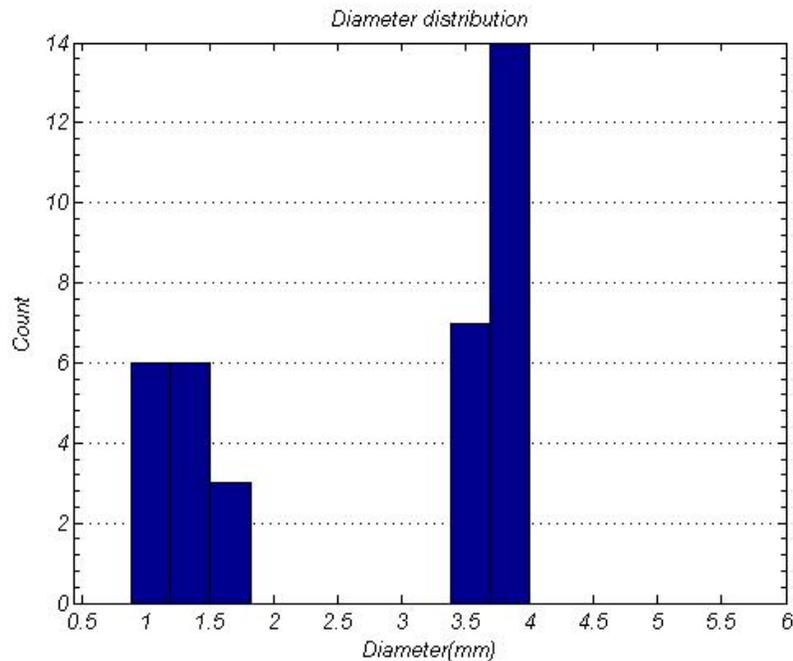


Figure 14. The diagram of particle size distribution for Sample 36 particles

It is confirmed that the identified and measured system only prescribes objects (particles). Thus, the algorithm for pattern recognition is proved to be proper for the present purpose. In the second part of the test, the mean area of particle surface ( $\text{mm}^2$ ) for the sample of the 36 particles is shown in Figure 15. In this test, the threshold level of intensity is chosen about 0.5 for better processing. It should be noted that the measured particle size depends on accurate value of number of pixels involving metric unit to the scale in the digital image.

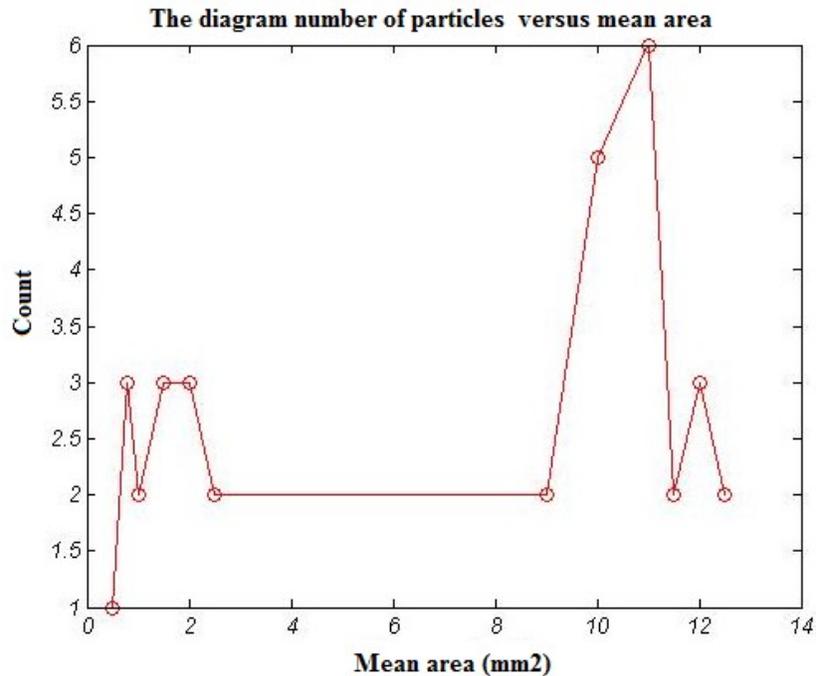


Figure 15. The diagram of particles mean cross-sectional area distribution for Sample 36 particles

The average size of particles, 2.7352 mm is calculated using program and the standard deviation is 1.2385mm which shows clearly that results of the automatic measurement are quite satisfactory. If the size of a pixel is increased (that is, if the photographs of the particles have lower magnification), results of the automatic measurement would be of lower accuracy due to maximum measurement uncertainty. Since tiny or small objects with lower resolution show weak results, it is essential to obtain high-quality and high-contrast photographs of the objects.

This method can be used as a simple pattern recognition technique programmed to size and count a relatively large number of particles in a short time and it is applicable to filter out images of undesirable objects such as redundant spots or noises in the image. Proper choices of the magnification in constructing an image frame and a threshold level to convert the original image to a binary digital image are important. Furthermore, due to light effects, it is observed that black color sheets are better than white color sheets for placing particles.

## 6. Measuring the Blood Cells by Means of an Image Segmentation

The analysis of the blood cells is used for evidencing several diseases in humans and animals. Usually, the analysis is performed manually by means of an optical microscope. However, some approaches, based on an automatic image processing, exist for evidencing and counting the blood cells in the given specimens. Measurements of the cell morphology are possible too. Here we show a method which uses a processing based on an image segmentation obtained by thresholding. After the segmentation, the area of the cells is easily evaluated. Their perimeters are measured using a rose of directions (Amelia, C., S. 2017).

Besides the counting of particles, from the micrographs that we can obtain by microscopes, some measurements concerning the shape of on cells are possible. In fact, the measurement of the size and size distribution of blood cells is the proper task for the method that we have recently developed and applied. To the counting of particles we can add some measurements concerning the cell morphology, measurements that we can obtain by means of a rose of directions. One of the measures is the perimeter of the cell, but it is possible to give several other parameters. In image processing, a segmentation is a method of partitioning an image into multiple sets of pixels, defined as super-pixels, in order to have a representation which can be simpler than the original one or more

useful to the following desired analyses. Specifically, the segmentation is a process of assigning a label to every pixel in an image, such that the pixels having the same label share certain characteristics. Several methods exist for the image segmentation, but here we use a method based on a thresholding of grey-scale images. By selecting a suitable threshold, the image is converted into a binary (black and white) image. In several cases, this is enough for evidencing the domains among the black or the white pixels. Therefore, here we start immediately to show the results of the segmentation of an image of blood cells. We show that we can measure their apparent areas and perimeters. Once we have obtained the binary image, we have at our disposal a matrix of black and white pixels. Starting from the left/upper corner of this matrix, we move on each row from left to right, from the upper to the lower row of the image. We focus on black pixels and characterize each of them by a sequential integer number  $k$ , which is acting as a label of the single pixel. Some of these labels will be the labels identifying the domains, or super-pixels, to which the pixels belong (see Figure 16).

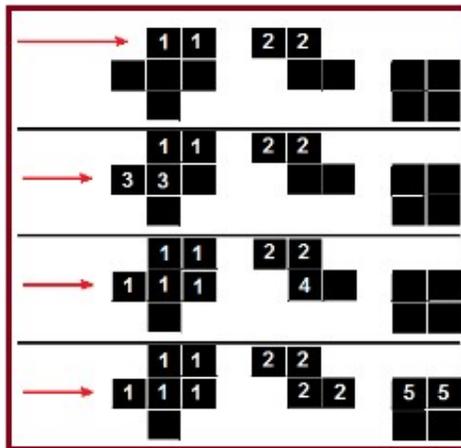


Figure 16. Labeling of the black objects including black pixels.

The white pixels have a label  $k=0$ . Each black pixel has a label different from zero, with a value which is increasing as we move from left to right on the rows. However, the final values are fixed according to the labels of the black pixels above and on the left of the considered pixel. The label of a black pixel is determined according to the labels  $k$  of the nearest black pixels above (A) and on the left (L) of the considered pixel. If the labels  $k_A$  and  $k_L$  are the same, their value is the label of the pixel under examination. If these labels are different, the pixel assumes as its label the lower value among them. Then, all the pixels having the label with the larger value change their labels into the value of the smaller one. This approach can be easily obtained by logic instructions in any programming language (here we use MATLAB 2016a). It gives to each of the super-pixels a different label. After this procedure, we have a matrix of labels. Suppose, that we want to segment the blood cells, so mark all the pixels belonging to the desired object. All pixels under the threshold may belong to the objects. We can determine the threshold level by  $\text{imagesc}(b < \text{threshold})$ ; in MATLAB programming and is between 0 to 255 for each pixel.

### 6.1. Counting and Sizing Human Blood Cells

Let us consider the image of human blood cells Figure 17 (a). The colour image is converted into a grey-scale image and then the binary image obtained manually by means of BMP image (Figure 17, b). As we can see, the white cell is not well defined in the binary image. For this reason, in this preliminary work, we modified manually the image to be used for segmentation. The modified image is given in the Figure 17 (c).

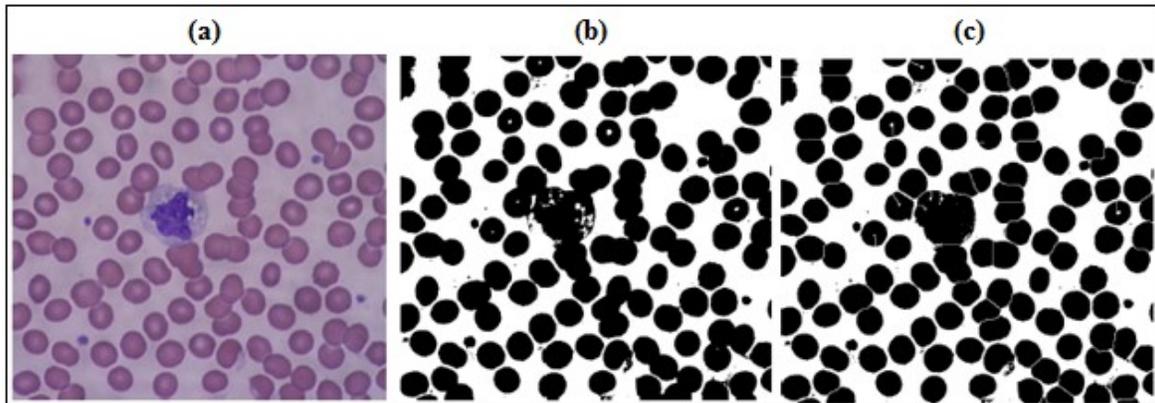


Figure 17. (a) Human blood cells magnified 40 times, (b) BMP image, (c) Modified image

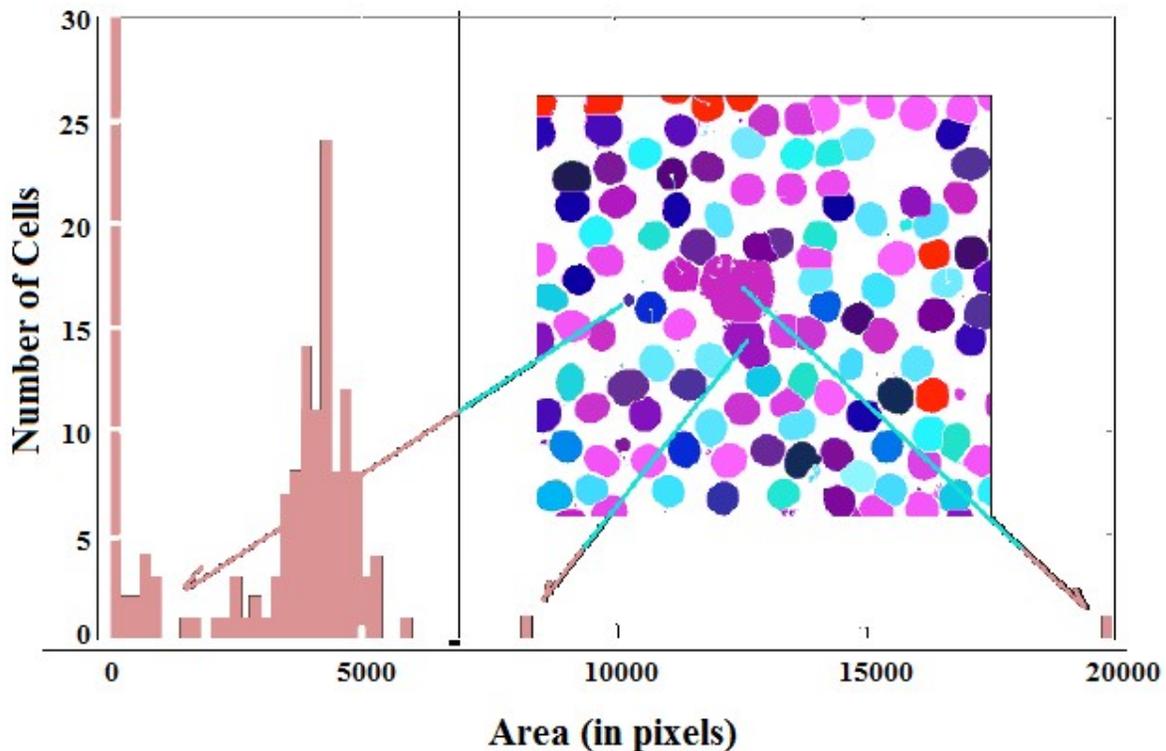


Figure 18. Segmentation and size distribution of the various white cell

Of course, we can measure the size and the perimeter of the cells too. In this preliminary work, in the case of the human blood, we needed to act manually to adjust the shape of the white cell. Further work is necessary to determine an automatic manner of improving the visibility of the white cells. An example of thresholding using generalized method is given in the Figure19. The result is quite interesting, because it is evidencing the white cells. By the segmentation of the Figure19, right panel, we can measure the morphology of the white cells. Another interesting result which it was obtained using a three-level image made of white, black and grey pixels. The results is reproduced in the Figure 20. Using only the black pixels, we have a map as given in the Figure20 (a). Using only the grey pixels, we have the image Figure20 (b), in the same figure. On these two images, the approach for having the cell morphology can be applied, after an ImageJ (Software) Watershed separation. An issue still exists for the red cells and it includes in the fact that some domains can have an empty. A method of processing the shape of the cells having an empty core is under development.

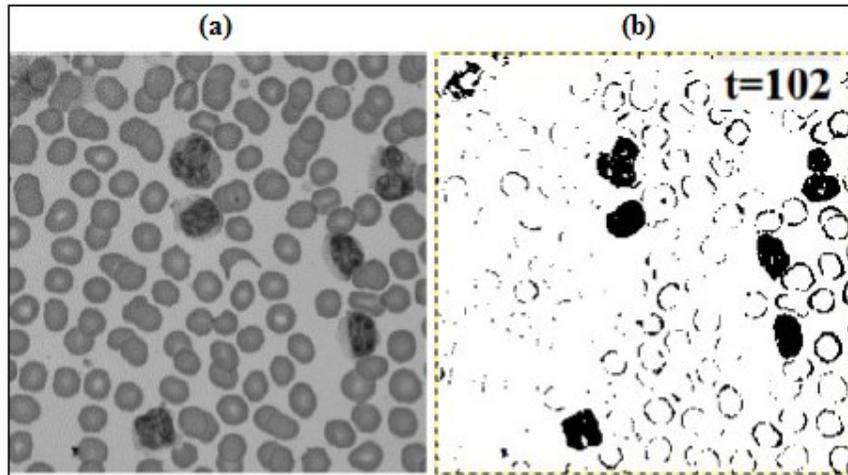


Figure 19. (a) Microscopic image of blood cells on the left. (b) Assuming the best bi-level image being that giving the largest number of edge pixels with threshold value of 102

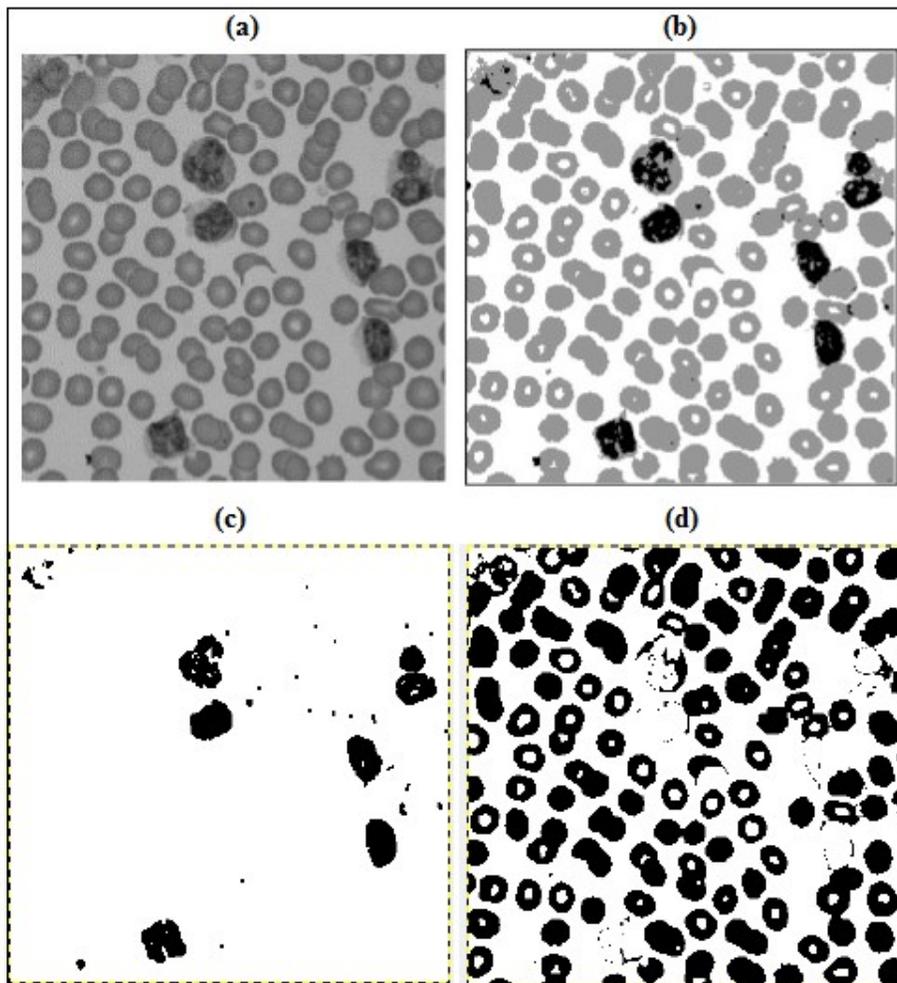


Figure 20. (a) Original and (b) Three-level images.  
(c) The black pixels and, (d) The grey pixels

We can apply the approach previously discussed for the determination of the cell morphology.

## 7. Conclusion

The research presents automated procedure to set together size and shape of particles to identify their boundaries in images and specify shape variations with shape group and size of the particles. The advantage of this algorithm is that it provides a tool to measure size of particles in digital images using MATLAB. This tool expedites the process of obtaining information of particles which helps optimizing the synthesis process of particles. Thus, an efficient and economic system for measuring particles' size is developed and tested. The system can handle a relatively large number of particles in a short time and can filter out images of undesirable objects such as noises, fine dots, spots and odd-shaped foreign materials in the background without complicated feedback processes. Size (diameters) of particles are obtained from the arithmetic mean value of the X-directional length of the longest continuous strip of pixels and the Y-directional length of the modified binary image of each object (particles). The test results show that the main error originates from converting the original images into binary images. Thus, the threshold level should be chosen carefully. The proper choice of magnification in constructing an image frame is another important consideration. In addition, for better image sensing, it is essential to obtain high-contrast photographs of particles. The image processing system described here, saves measuring time, eliminates possible subjective errors by observers in sizing and counting particles, considerably and is shown to be substantially efficient. It can be understood that black color sheets are better than white color sheets for placing particles. Shadow effects on black color sheets should be less than that of white color sheets. Also, the result of a segmentation is a set of "segments", or "super-pixels", or "domains" that are covering the background of the whole image. The segmentation can also give a set of contours that is of "edges", extracted from the image. In this case, the segmentation gives an "edge detection". In addition, results of this method are more accurate for larger object and high-resolution images.

## References

- Ahler, K. D., & Alexander, D. R. (1985). Microcomputer based digital image processing system developed to count and size laser-generated small particle images. *Optical Engineering*, 24(6), 241060.
- Amelia, C., S. (2017). Measuring the blood cells by means of an image segmentation. *Philica*, Banta, L., Cheng, K., & Zaniewski, J. (2003). Estimation of limestone particle mass from 2D images. *Powder technology*, 132(2-3), 184-189.
- Castleman, K. R. (1979). Stereometric Ranging. *Digital Image Processing*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 364-369.
- Coster, M., & Chermant, J. L. (2001). Image analysis and mathematical morphology for civil engineering materials. *Cement and Concrete Composites*, 23(2-3), 133-151.
- Fernlund, J. M. (1998). The effect of particle form on sieve analysis: a test by image analysis. *Engineering Geology*, 50(1-2), 111-124.
- Fernlund, J. M. R. (2005). Image analysis method for determining 3-D size distribution of coarse aggregates. *Bulletin of Engineering Geology and the Environment*, 64(2), 159-166.
- Fishler, R., Verhoeven, F., de Kruijf, W., & Sznitman, J. (2018). Particle sizing of pharmaceutical aerosols via direct imaging of particle settling velocities. *European Journal of Pharmaceutical Sciences*, 113, 152-158.
- Gontard, L. C., Ozkaya, D., & Dunin-Borkowski, R. E. (2011). A simple algorithm for measuring particle size distributions on an uneven background from TEM images. *Ultramicroscopy*, 111(2), 101-106.
- Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2004). *Digital Image Processing Using MATLAB*, Prentice Hall. Upper Saddle River, NJ.
- Hahn, B., & Valentine, D. T. (2016). *Essential MATLAB for engineers and scientists*. Academic Press.
- <https://www.mathworks.com/discovery/digital-image-processing.html>

- Kim, I. G., & Lee, S. Y. (1990). A simple technique for sizing and counting spray drops using digital image processing. *Experimental thermal and fluid science*, 3(2), 214-221.
- Kumara, G. H. A. J. J., Hayano, K., & Ogiwara, K. (2011). Fundamental study on particle size distribution of coarse materials by image analysis. In *First International Conference on Geotechnique, Construction Materials and Environment*(pp. 399-404).
- Kumara, G. H. A., Hayano, K., & Ogiwara, K. (2012). Image analysis techniques on evaluation of particle size distribution of gravel. *Int. J. Geomate*, 3(1), 290-297.
- Maiti, A., Chakravarty, D., Biswas, K., & Halder, A. (2017). Development of a mass model in estimating weight-wise particle size distribution using digital image processing. *International Journal of Mining Science and Technology*, 27(3), 435-443.
- Ow, C. S. (1981). Pattern recognition procedures for a television-minicomputer spray droplet sizing system. *Journal of the Institute of Energy*, 9, 119-123.
- Park, D. Y. (2018). Direct calculation of inter-particle distance in suspension by image processing. *Powder Technology*, 330, 252-258.  
*Philica, Available HAL, hal-01654006.*
- Phromsuwan, U., Sirisathitkul, C., Sirisathitkul, Y., Uyyanonvara, B., & Muneesawang, P. (2013). Application of image processing to determine size distribution of magnetic nanoparticles. *Journal of Magnetism*, 18(3), 311-316.
- Pratt, W. K. (1978). *Digital Picture Processing*.
- Rajeshwari, A., Prathna, T. C., Balajee, J., Chandrasekaran, N., Mandal, A. B., & Mukherjee, A. (2013). Computational approach for particle size measurement of silver nanoparticle from electron microscopic image. *Int. J. Pharm. Pharm. Sci*, 5(2), 619.
- Reetz, M. T., Maase, M., Schilling, T., & Tesche, B. (2000). Computer image processing of transmission electron micrograph pictures as a fast and reliable tool to analyze the size of nanoparticles. *The Journal of Physical Chemistry B*, 104(37), 8779-8781.
- Weiss, B. A., Derov, P., DeBiase, D., & Simmons, H. C. (1984). Fluid particle sizing using a fully automated optical imaging system. *Optical Engineering*, 23(5), 2355-61.