An Autonomous Program for Crack Length Calculation In An Unsaturated Soil In 1-D Column

Anangsha Alammyan[1[0000-0003-0649-2950]], Sai Krishna Kothapalli[1[0000-0002-5820-9427]] and Sreedeep S[1[0000-0003-2566-6901]]

[1] Indian Institute of Technology Guwahati, Guwahati- 781039, Assam, India
anangsha.nits@gmail.com, kothapalli@iitg.ac.in, srees@iitg.ac.in

Abstract. Cracks are widely present in natural and engineered soils. Their presence has the potential to significantly alter both hydraulic and mechanical properties of soil, thereby causing the performance to weaken with respect to different engineering disciplines, particularly geotechnical, environmental and geological engineering. This makes the quantification and characterization of cracks important, which would lead to a better understanding about their extent and complexity in soil under various conditions of seasonal rainfall changes. The length of individual cracks has been studied in literature to determine the smallest volume with respect to which an averaging of a certain property of the cracked soil can be performed. In this regard, the analysis of images has been instrumental in crack length quantification. However, the currently-used software-based methods involving manual adjustment through visual inspection of the cracks in 2-dimesional surface images is a time-consuming task that lacks accuracy, sensitivity and reproducibility. There is, hence, the need to develop automated imaging techniques for analyzing crack length in drying soils rather than relying on software-based methods. The main objective of this study is to introduce a novel image analysis tool that employs an open source computer vision library coded in Python to effectively characterize crack length in soils. A simple experimental set-up was developed using a 1-dimensional column containing compacted red soil in an environment-controlled chamber. A series of images of soil sample was captured using commercially available camera model (Canon EOS 700D) to have a photographic time-lapse representation of the cracking process. A step-by-step strategy using a Python script is presented here to outline an approach to divide the circular 2-dimensional image into equal radial slices and calculate the crack length in each slice. The approach can be further extended to calculate other crack parameters sector-wise in a circular image.

Keywords: Surface cracks, Python code, autonomous, crack length density

1 Introduction

Upon drying, shrinkage might occur in soils due to loss of water by evaporation. A desiccation crack generally appears in soil when the volumetric shrinkage is constrained or when the induced surface tensile stress becomes equal to the soil tensile strength [1]. The presence of desiccation cracks in soil can fundamentally modify both its hydraulic and mechanical properties, thereby causing a weakened performance of soil. The stability of geotechnical structures that have been constructed on clayey soils is significantly affected directly or indirectly by the presence of desiccation cracks in the soil matrix. Study of desiccation crack is important while considering urban green infrastructure like terrace gardens, bare soil infrastructure like embankments, bio-engineered slopes and agricultural field.

Several field and laboratory studies have been carried out for the characterization of desiccation cracks in soils using variety of approaches like manual measurement [2], X-Ray computerized tomography [3], electrical resistivity tomography [4], and numerical simulation [5] among many others. The analysis of digital images of the soil surface has been extensively used to get an idea of the two-dimensional crack characteristics of soil [6,7]. As distribution and extent of crack network affect water storage and movement, it is important to measure the crack size and pattern. The determination of crack geometry is also important in characterizing the diverse phenomena associated with cracked soils. Studies have stressed on the need for an accurate description of spatial characteristics of joint systems and desiccation cracks [8]. Knowledge of the spatial characteristics of cracks is helpful in environmental applications to accurately model dispersion of contaminants and porous flow. In previously conducted laboratory studies, a commonly used parameter called the crack length density (CLD) has been defined as the total length of the crack skeleton in a soil mass. The calculation of crack length density involves computation by accumulating pixel lengths between two adjacent skeletal pixels. It has been studied to determine the crack geometry [9,10] and to characterize the smallest volume for which a specific property of the cracked soil can be averaged over [11].

In previous studies, the computation of crack length was done by mapping the entire soil area using AutoCAD and calculating the lengths of individual cracks by approximating them as straight lines [9,11]. There appears to be a lack of automation in the reported studies. In addition, a significant amount of computational time and human effort need to be spent in analyzing each image separately. This has significant impact in case of studies covering large areas which generate several photographs for which the CLD has to be determined. Moreover, the chances of reproducibility in the results from manual processing is less because of the effect of observer-dependent subjectivity. The main objective of the present study is to develop a program to automate the calculation of CLD of surface cracks in unsaturated soil in a one-dimensional (1-D) column. It aims to overcome the shortcomings of the existing methods of determination of the CLD by using a program coded in Python language. The Python code analyzes 2-dimensional photographs of cracked soil samples and gives the output in the form of a binary image showing the crack skeleton, from which the CLD can be determined. It is an effective image analysis technique valid for all types of soils tested in the laboratory using test pots of any size.

2 Methodology

2.1 Test plan, setup and instrumentation

Soil properties. The tested soil was collected from a hill site in Guwahati, India. The index properties of the soil were determined following the standard procedures given by ASTM [12,13,14]. The sand, silt and clay contents were determined to be 26%, 50% and 24%, respectively. Liquid and plastic limit were found to be 42% and 24%, respectively. According to the unified soil classification system (USCS), the soil was classified as inorganic silt of low plasticity (ML). The maximum dry density (MDD) was determined to be 1690 kg/m^3 with corresponding optimum moisture content (OMC) of 17%. The other index properties of the tested soil are listed in Table 1.

Table 1. Index properties of the tested soil.

Index properties	Value
Specific gravity	2.58
Particle-size distribution (%)	
Coarse sand (4.75 mm–2 mm)	0
Medium sand (2 mm–0.425 mm)	7
Fine sand (0.425 mm–0.075 mm)	19
Silt (0.075 mm–0.002 mm)	50
Clay (<0.002 mm)	24
Specific gravity	2.58
Atterberg limit (%)	
Plastic limit	24
Liquid limit	42
Plasticity index	18
Shrinkage index	21.3
Standard compaction tests	
Maximum dry density (kg/m^3)	1690
Optimum moisture content (%)	17

Sample preparation and testing procedures. The soil was first oven-dried. The soil samples were placed in a Poly Vinyl Chloride (PVC) column with a height of 250 mm and an inner diameter of 300 mm. The soil column was placed on a perforated base plate, over which a filter paper was placed. This was done to prevent loss of soil particles with seeping water. All the soil samples were compacted statically to 0.9 Maximum dry density (MDD), which is commonly used in embankment soil [15]. A thin layer of lubricant was applied on the inner surfaces of the column before compaction to reduce the soil-PVC interface friction. The compaction procedure was divided into three layers, each having an equal thickness to achieve uniform soil density. Three replicates (in three separate molds) were prepared to carry out the time-lapse study. The surface of the sample was leveled using a straight edge to get a uniform surface. This procedure was repeated for all the specimens. After compaction, the entire column was placed inside a temperature and relative humidity (RH) controlled chamber where the temperature (25 ±2 ^{0}C) and RH (40 ±5 %) were controlled to minimize extreme environment variations. Ultraviolet lighting was turned on to simulate natural lighting conditions. This is shown in Fig. 1. During the process, the soil samples were monitored continuously for a period of 28 days.

Monitoring of cracks. During the experiment, continuous monitoring was done using a high-resolution digital still camera placed straight above the sample. The images were taken with a digital camera Canon EOS 700D with an exposure time of 1/50 second, ISO speed 160 and 39mm focal length of lens. To get undistorted images directly above the testing specimen, the camera was placed on an adjustable steel mount above the sample. During the duration of the experiment, the camera settings remained fixed. The surface size of the three specimens used for the purpose of image analysis was taken as to be of diameter 300mm in accordance to a scale-effect study carried out to find the representative elementary volume [11].

Fig. 1. Test set-up for monitoring crack initiation and propagation by image analysis

2.2 Image Analysis.

Image analysis is a simple non-invasive technique of analyzing digital images of cracked soil surface for the characterization and quantification of desiccation cracks. The developed cracks have a distinct color which is significantly darker than the soil matrix. This assists in the identification of crack pixels and their distinction from the intact soil area. The present study puts forward a novel approach for measuring the lengths of all surface cracks in a cylindrical soil sample by analyzing the digital cracks using a script coded in Python. The thresholding is done with the aid of a novel image analysis algorithm using a step-by-step strategy with a script coded in Python [16]. The first step is the conversion of the original RGB image to greyscale which effectively removes all color information while retaining information on light intensity. The applied method employs the BGR2GRAY command in Python that converts the RGB image to grayscale. The second step uses a smoothing operation to blur out the surface irregularities. This is done by applying a bilateral filter, which removes irregularities without blurring out the crack edges. The next step is thresholding the original image to form a binary image where the crack skeleton appears in black and the soil matrix appears white. This is done using the Gaussian adaptive thresholding method. In this step, each pixel in the greyscale image is replaced with a black pixel if the image intensity is less than a threshold value, and it is replaced with a white pixel if the image intensity is greater than that particular threshold. The value of the threshold for a particular pixel is automatically selected depending upon the light intensity of the surrounding pixels.

The thresholded images then are divided into eight equal sectors to find the CLD in each sector. Summing up, the total crack length in the image can also be found. There were some challenges faced while developing this algorithm. Each image was clicked manually, and hence the resulting circular soil sample had varying diameter and different dimension of shadow from the container enclosing it. Because of this, it was difficult to locate the center of each different circle with a single code for all images. Keeping the above difficulties in mind, the authors came up with an algorithm where a sector was created of specified angle of 45º. This individual sector was masked over the circular thresholded image of the soil sample and then rotated around it. With this method, the values of crack lengths could be found out sector-wise for the entire image. A Python script was written to incorporate the same. The procedure is explained in brief in the following steps.

At first, the image is thresholded as per a novel algorithm [16]. Following which, a sector having an angle of 45º is drawn and masked on the thresholded image of the cracked soil. The rest of the image is made transparent, so, when the program runs, the analysis is carried out only on the selected area, which is one sector of 45º. The total length of the cracks in that particular sector is then measured by counting the number of black pixels in it. The novel script does this automatically and no human input is required once the program starts running. After the measurement of crack length is done in the first sector, the original masked sector is rotated by 45º till the next portion of the image is covered. This leads to the formation of a second sector in which measurements can be carried out. The rest of the image including the first sector is made transparent to ensure that measurements are taken only in the selected area. After the crack length in the second sector has been calculated, the masked sector is further rotated by another 45º so that a third sector in the image is selected and the rest of the image is rendered transparent. After the measurements have been made, these steps are repeated until the whole image is covered by the masked sector leading to the original image of the circular soil sample being effectively divided into eight equal sectors of 45º each. Thus the crack length density for each sector can be found separately while the rest of the image has been taken out of consideration for calculation. As a summation of all the above, the total crack length in the image can be found. An additional parameter called the crack intensity factor (CIF) defined as the ratio of area of cracks to the total area of soil sample can be determined as the ratio of black pixels to white pixels in the thresholded image. CIF gives an idea of what percentage of the soil surface is cracked. The entire process with the corresponding segment of code is summarized in Fig. 2.

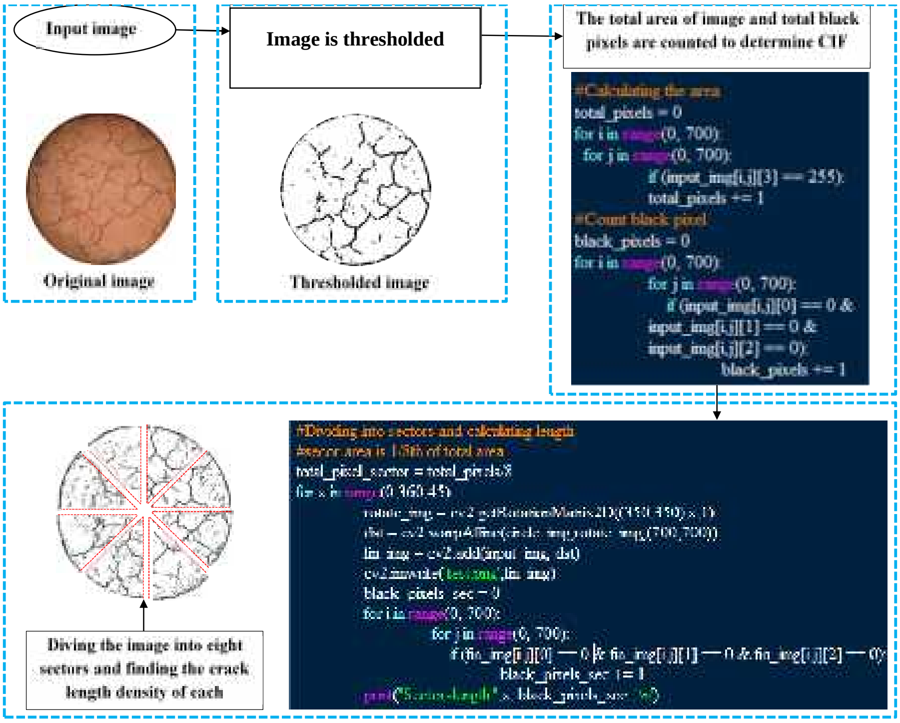

Fig. 2. Flowchart outlining the steps involved in splitting the image to sectors and calculating the crack length density

3 Results and discussion

The scope of this novel approach is limited to the measurement of crack length by counting the number of dark pixels as the cracks in the thresholded image appear as dark while the surrounding soil sample appears as white. Further studies can be carried out to use this masking algorithm to divide the circular images of soil samples into equal sectors and find out other crack parameters in each sector. Previous studies have been carried out where the entire circle was divided into an orthogonal network of square grids, along which, the crack parameters are counted [17]. This method of dividing the image into smaller areas leads to possible errors in estimation of crack parameters as the periphery of the image is curved and hence is not covered in full square grids. The Python script presented here for masking the image can be considered as a suggestion for improvement of this existing method. Dividing the image into radial slices with an equal angle rather than squares could be a more uniform way of dividing the image to estimate crack parameters. This could also save computational time otherwise in accounting for the discrepancies along the circumference of the circle in case of orthogonal grids. The results from the study calculated using the algorithm is shown in Table 2.

Table 2. Specimen geometry and main results from image analysis

Total Area (cm^2)	**Area of uncracked Soil (cm^2)**	**Total Crack Area (cm^2)**	**CIF**	**Crack Length Density (cm)**	**Crack length/ unit area (cm^{-1})**
706.86	537.21	169.65	0.24	122.48	0.173

4 Conclusion

The study presented a simple autonomous program customized for quantifying the total length of surface cracks in unsaturated soil in a 1-D column. The program efficiency to compute CLD was validated using 1-D column experiments. Though, a simple program was introduced for quantifying cracks in 1-D column, however, the same can be easily modified for use in different set ups. The authors would like to provide copy of the program as open source for allowing users to further modify it.

4.1 Scope for future work. Image analysis has proved to be a powerful tool for the qualitative and quantitative description of 2D patterns. The experimental set-up and image analysis described in this paper is part of a long-term project investigating the cracking patterns of drying soils. Applications of the method described here to study the fundamentals of soil cracking and changes in suction can be found in related publications [16]. Research suggests that cracking is a three-dimensional phenomenon [18] and the 3-dimensional quantification of crack patterns is a much more difficult task than the 2D process described in the current work. In addition to advanced techniques, the study requires more sophisticated equipment. In essence, the novel method described here can be considered as a first step toward this ultimate goal of quantification of crack networks in the field and in geotechnical projects.

References

1. Corte, A. and Higashi, A.: Experimental Research on Desiccation Cracks. Soil-Research Report, 66 (1960).
2. Ackerson, J.P., McInnes, K.J., Morgan, C.L. and Everett, M.E.: Measuring Crack Porosity using Three-Dimensional Electrical Resistivity Tomography. Soil Science Society of America Journal, 81, 1025-1035 (2017).
3. Zhou, H., Mooney, S.J. and Peng, X.: Bimodal soil pore structure investigated by a combined soil water retention curve and X-ray Computed Tomography approach. Soil Science Society of America Journal, 81(6), pp.1270-1278 (2017).
4. Samouëlian, A., Cousin, I., Richard, G. and Tabbagh, A.: Soil cracks detection by 3D electrical resistivity. In 9th EAGE/EEGS MEETING. Springer, Kyiv, Ukraine (2003).

5. Amarasiri, A., Kodikara, J., Costa, S.: Numerical Modelling of Desiccation Cracking. International Journal for Numerical and Analytical Methods in Geomechanics 48, 336–340 (2010).
6. Yoshida, S. and Adachi, K.: Effects of cropping and puddling practices on the cracking patterns in paddy fields. Soil Science and Plant Nutrition, 47, 519-532 (2001).
7. Peng, X., Zhang, Z.B., Gan, L. and Yoshida, S.: Linking Soil Shrinkage Behavior and Cracking in Two Paddy Soils as Affected by Wetting and Drying Cycles. Soil Science Society of America Journal, 80, 1145-1156 (2016).
8. Bosscher, P. J. and E. C. Douglas.: Measurement and Analysis of Jointing Properties in Fine-Grained Soils. Journal of Geotechnical Engineering, ASCE 114(7):826-843 (1988).
9. Zhang, Z.B., Peng, X., Wang, L.L., Zhao, Q.G. and Lin, H.: Temporal changes in shrinkage behavior of two paddy soils under alternative flooding and drying cycles and its consequence on percolation. Geoderma, 192, 12-20 (2013).
10. Wang, C., Zhang, Z.Y., Liu, Y. and Fan, S.M.: Geometric and fractal analysis of dynamic cracking patterns subjected to wetting-drying cycles. Soil and Tillage Research, 170, 1-13 (2017).
11. Li, J.H. and Zhang, L.M.: Geometric parameters and REV of a crack network in soil. Computers and Geotechnics, 37, 466-475 (2010).
12. ASTM. “Standard test method for particle-size analysis of soils”. ASTM International, West Conshohocken (2007).
13. ASTM. “Standard Test Methods for Liquid Limit, Plastic Limit, and Plasticity Index of Soils”, ASTM International, West Conshohocken, PA (2010).
14. ASTM. “Standard Test Methods for Laboratory Compaction Characteristics of Soil Using Modified Effort”. Annual Book of Standards 4 (2017).
15. Li, J.H., Li, L., Chen, R. and Li, D.Q.: Cracking and vertical preferential flow through landfill clay liners. Engineering Geology, 206, 33-41 (2016).
16. A. Anangsha, V. K. Gadi, S. Bordoloi, S. K. Kothapalli, S. Sreedeep, M. Guoxiong, and A. Garg: A New Autonomous Program Customized for Computing Surface Cracks in an Unsaturated Soil in a 1-D Column. Journal of Testing and Evaluation 47(5), 3822–3835 (2019).
17. Chaduvula, U., Viswanadham, B.V.S. and Kodikara, J.: A study on desiccation cracking behavior of polyester fiber-reinforced expansive clay. Applied Clay Science (2017).
18. Zhang, Z.B., Zhou, H., Zhao, Q.G., Lin, H. and Peng, X.: Characteristics of cracks in two paddy soils and their impacts on preferential flow. Geoderma, 228, 114-121 (2014).

APPENDIX: Python script for quantifying the CIF.

```
import numpy as np
import cv2
from matplotlib import pyplot as plt
import sys
from math import pi

if int(len(sys.argv)) < 2:
    print ("Usage :python script.py input.png output.png")
    sys.exit()
```

```
# Read the Input Image
input_img = cv2.imread(sys.argv[1],cv2.IMREAD_UNCHANGED)
height, width, channels = input_img.shape
#Resizing the Input Image
input_img = cv2.resize(input_img, (700, 700))
#Splitting the channels
(inp_r, inp_g, inp_b, inp_a) = cv2.split(input_img)
#Convert to Grayscale
input_img = cv2.cvtColor(input_img, cv2.COLOR_BGR2GRAY)
#Blur it a bit to remove noise
input_img = cv2.bilateralFilter(input_img,5,75,75)
#Threshold the image using adaptive gaussian
in-
put_img=cv2.adaptiveThreshold(input_img,255,cv2.ADAPTIVE_THRESH_GAUSS
IAN_C,cv2.THRESH_BINARY,11,7)
#Find Contours
input_img, contours, hierarchy = cv2.findContours(input_img, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
#Draw Contours
input_img = cv2.drawContours(input_img, contours, -1, (0,255,0), 1)
#Adding the Alpha channel back
input_img = cv2.merge((input_img,input_img,input_img,inp_a))
#Save the image
cv2.imwrite(sys.argv[2],input_img)

#Calculating the area PI*radius*radius
total_pixels = 0
for i in range(0, 700):
    for j in range(0, 700):
        if (input_img[i,j][3] == 255):
            total_pixels += 1

#Count black pixel
black_pixels = 0
for i in range(0, 700):
    for j in range(0, 700):
        if (input_img[i,j][0] == 0 & input_img[i,j][1] == 0 & input_img[i,j][2] == 0):
            black_pixels += 1

print ("Total Number Of Pixels: ", total_pixels)
print("Total Black Pixels: ", black_pixels)
print ("CIF:", black_pixels/total_pixels)

#Dividing circle into sectors and calculating CLD
```

```
total_pixel_sector=totalpixels/8
for x in range(0,360,45):
        rotate_img=cv2.getRotationMatrix2D((350,350),x,1)
        dst=cv2.warpAffine((circle_img,rotate_img,(700,700))
        fin_img=cv2.add(input_img,dst)
        cv2.imwrite(‘test.png’,fin_img)
black_pixels_sec=0
for i in range(0,700):
   for j in range (0,700):
       if (fin_img[i,j][0]==0 &  fin_img[i,j][1]==0 & fin_img[I,j][2]==0):
      black_pixels_sec += 1
print(“Sector Length”,x, black_pixels_sec,‘\n’)
```