2025 ASEE Northeast Section Conference, Mar. 22, 2025, University of Bridgeport, Bridgeport, CT, USA.

A color-based image analysis tool used for engineering education and its potential application for AI training

Priscilla C. Fonseca
School of Computing and Engineering
Quinnipiac University
Hamden, CT USA
pcfonseca@quinnipiac.edu

Riley Popp
Cockrell School of Engineering
University of Texas
Austin, TX USA
rwp632@my.utexas.edu

***Abstract*—If a civil engineering structure fails, an investigation is usually launched to determine the cause of failure, which can include an assessment of the materials used. For example, if a concrete structure collapses, samples of the concrete are taken and examined through a petrographic analysis. A petrographer traditionally requires the use of sophisticated microscopes and costly proprietary software programs, placing a high barrier to entry and difficulty in incorporating the associated topics in the classroom and educational laboratory. To overcome this barrier, an algorithm and user-friendly Matlab application was developed to examine and quantify the constituent phases in samples of concrete. This tool performs color-based segmentation of the phases of concrete, including the computation of the area fraction of each phase. Since users of the application provide a wealth of information about each image, the app is well-positioned for incorporation into AI training and further improvements. The majority of the functions used in this app are open, and so students can better understand how the tool works, subsequently tinker with the tool, and hopefully build stronger understandings with coding and AI. This app can be incorporated into engineering courses and research on structural failure analysis and material condition assessment.**

***Keywords*—Image analysis; Segmentation; Petrography; AI training; Matlab**

I. Introduction

Image segmentation is a component of digital image processing that aims to group regions of an image according to similar characteristics, such as color, brightness, shape, or spatial properties. Segmentation is used in a variety of practical applications such as medical imaging, object detection, surveillance, and petrography, including concrete petrography. Concrete petrography is commonly used to evaluate the quality of a concrete sample, which may be needed in a forensic analysis. In this type of analysis, various properties are determined, such as the quantity and distribution of aggregates and air voids, the cement paste content, and damage indicators such as cracking and the presence of internal swelling around aggregates [1], [2]. Traditionally, a petrographer uses a stereomicroscope to examine a material sample surface by manually tracking the location of flaws and voids. The data is collected and analyzed using principles of stereology. This process can be tedious and slow. As an alternative, a flatbed scanner may be used for image capture. Flatbed scanners are relatively accessible because of its availability and relatively low cost. The image of a specimen obtained using a flatbed scanner can then be used to perform a digital image analysis, which is then used to examine the properties and health of a specimen.

Digital image analysis of concrete has been performed a number of times [3], [4], [5], [6], but the underlying code is often unavailable, or in situations of freely available code, multiple software packages are usually needed to achieve the desired results. For example, Multispec [7], [8], can be used with the ECHO Classifier algorithm to digitally segment the phases in concrete. However, when analyzing concrete, Multispec is typically used in conjunction with other image processing and analysis tools, such as ImageJ [9], forcing the user to shift between various tools and platforms. For researchers performing image analysis on concrete, the availability of Multispec and ImageJ has been a valuable resource; however, the code libraries can be challenging to run or customize, making this process less accessible to an undergraduate engineering student.

In recent years, advances in artificial intelligence (AI) have gained momentum in visual image analysis [10]. Notably, the convolutional neural network (CNN) is a robust approach used for visual understanding [11]. The application of CNNs has been successful in a number of computer vision tasks such as cancer detection [12], facial recognition, and autonomous driving. More recently, a CNN model was used to illustrate the feasibility of using AI training for concrete petrography [13]. This work also combined the use of Multispec for image segmentation, which was then used in the model training.

Prior work by the first author focused on analysis of air void systems of concrete and mortar specimens by thresholding grayscale images, which were then analyzed as a binary system [14]. We have expanded on this work by developing a color-based image analysis application which performs segmentation of images of multiphase materials. An interface was developed in Matlab that prompts the user to select a subset of regions from an image in order to classify all pixels that belong to that region. The process is repeated until the image is segmented into distinguishable phases along with a calculation of the area fraction of each phase. The user interface provides further options, refinement, and verification using an image masking feature. The application further allows the user to compare the results using the established k-means image clustering algorithm [15]. The proposed application can be used to analyze a variety of multi-phase materials. We present a multi-phase test case and several two-phase images of concrete. The package of Matlab functions are provided as supplementary material that can be downloaded, opened, and modified.

II. Methods

A. Algorithm

When a color image is captured using a scanner or digital camera, the device typically uses the RGB color model. RGB, which stands for red, green, and blue values, are three channels that are combined to produce a specific color. Alternatively, images may be represented using the hue, saturation, value (HSV) color model, which describes the colors in terms of saturation and brightness. The HSV colorspace can be converted from RGB colorspace using a straightforward algorithm [16].

Given an image of a multiphase material with each phase distinguishable by color, the following algorithm was written to segment the phases as follows:

1) The user selects a trial region, either by drawing a rectangle or making a series of point selections. The user attempts to select only pixels belonging to a single phase.
2) The RGB values of the selected pixels are tabulated. For each channel in the RGB color space, the minimum and maximum values are stored in order to determine the range of values in each channel. Alternatively, the same process is performed using the HSV color space.
3) Each pixel from the image is evaluated to determine if the color value in each of its three channels falls within the range computed in the previous step. If true, the pixel is classified as a foreground pixel.
4) The foreground pixels are classified together as a single phase and its area fraction is computed.

An application with a user interface was developed in order to provide various options and visual confirmation of the quality of segmentation. The features of the user interface are described in Section II-E.

B. Test case

The algorithm described in Section II-A was tested using a simple example prepared with Crayola markers and 1/2-inch graph paper. The sample was created by freehand coloring the squares on the paper using gray, black, brown, green, blue, and red, representing a 6-phase sample as shown in Figure 1(a). The sample was scanned using an Epson V800 flatbed scanner at 600 dpi resolution. Since each square was approximately the same size, and the area of the sample was known, the predicted area fraction of each color was computed easily. The algorithm was used to successfully segment and compute the area fractions of each of the six colors as shown in Figures 1(b)-(g) and Table I. An additional image showing unclassified pixels is included in Figure 1(h). The test was designed to capture physically generated colors, rather than digital colors on a computer. In this case, nearly all of the unclassified pixels fell at the boundaries of the squares where the markers naturally overlapped a small amount.

TABLE I
Fraction of each color of the test sample shown in Figure 1.

Color	Predicted area fraction	Segmented area fraction	Error
Gray	1/16 = 6.25%	6.4%	2.4%
Black	1/16 = 6.25%	6.3%	0.8%
Brown	2/16 = 12.5%	12.7%	1.6%
Green	3/16 = 18.75%	18.1%	3.5%
Blue	4/16 = 25%	25.1%	0.4%
Red	5/16 = 31.25%	30.9%	1.1%
Unclassified	–	0.5%	–

C. Sample preparation

Concrete samples were cast and wet-cured into 2-inch cube molds. All mixtures contained Type I Portland cement, course and fine aggregates, and an air-entraining agent. Each sample was cut with a diamond masonry saw and lapped with a Buehler grinder/polisher using successively finer silicon carbide grinding paper up to 800 grit. If segmentation of the cement paste was needed, the surface was sprayed with a 1% phenolphthalein pH indicator solution, which stains alkaline materials a purple color. After 5 minutes of phenolphthalein exposure, the surfaces were wiped with a cloth and placed in a 105°C oven for 5 minutes to dry. If the cement paste was not to be segmented, then the sample surface was left unstained.

If air voids were to be segmented, a 1-2 μm powder manufactured by Stardust Micas was pressed into the sample surface and then scraped with a flat spatula. A variety of chalk colors were tested, including green, orange, yellow, and blue, but the best images were produced using yellow chalk on pink/purple phenolphthalein-stained cement or orange chalk on unstained gray cement.

D. Image capture

Each sample was scanned in tif file format as an RGB image using an Epson V800 flatbed scanner. If air void analysis was desired, the image was taken at 4800 dpi resolution, corresponding to a 5.29-μm pixel size, which was the maximum optical resolution of the scanner. However, because the 4800 dpi image file sizes were quite large, depending on the size of the smallest features, a lower resolution, such as 600 dpi (42.3-μm pixel size), was used. The images were not rotated or enhanced, but the edges of the images were trimmed in order to remove the background color and edge effects.

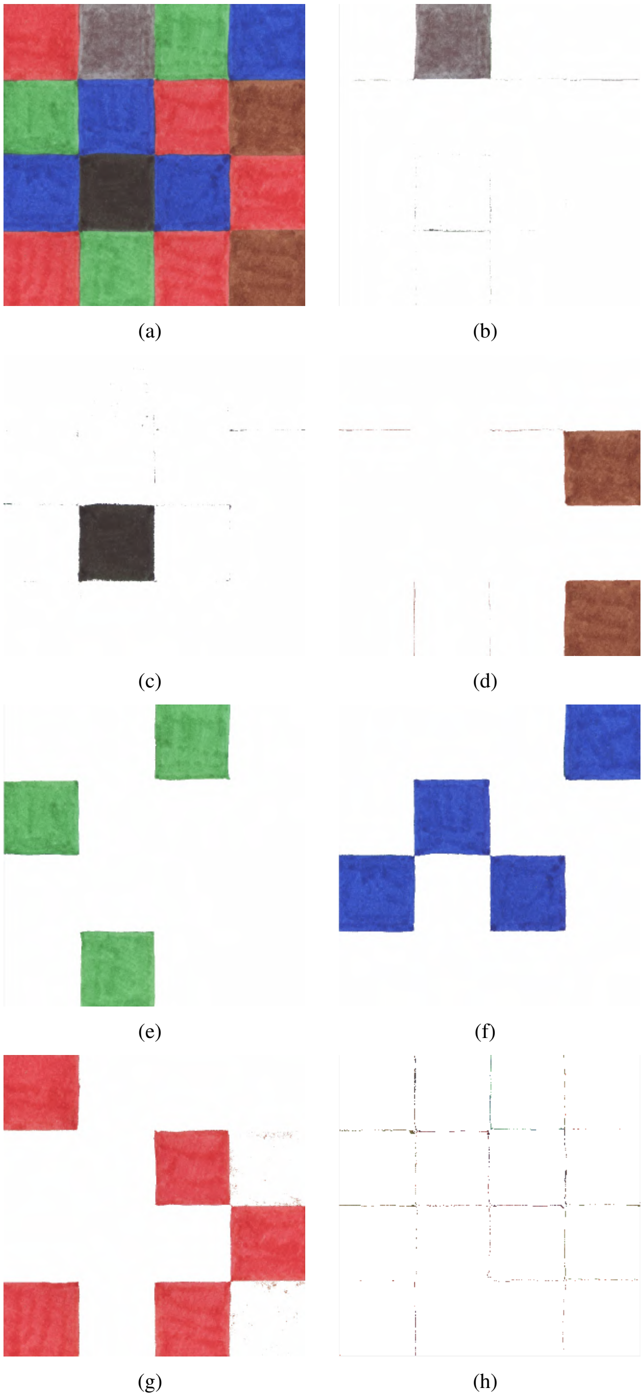


Fig. 1. Results of a 2-in x 2-in test sampled created using grid paper and color markers: (a) Original image, segmented (b) gray, (c) black, (d) brown, (e), green, (f) blue, (g) red, and (h) unclassified pixels.

E. User Interface

When executing the application, the user is prompted to select a truecolor image, and then enter the number of phases present. The image will display with a menu as shown in Figure 2. Zoom, pan, data tips, and other options are available by hovering the mouse at the top right corner of the figure.

The first drop down menu contains the options **Original image** or **Background**, which tells the program whether the original image or the background of the working phase will be used to select a trial region. At the start of the program, the user should use the Original image option. After running the first mask, the user can then choose to select new trial regions using either the original image or the background image.

When **Draw new region** is selected, the user draws a blue rectangular region onto the image corresponding to the working phase. The rectangle may be adjusted in size. The **Select** button is pressed in order to confirm the selection.

When **Place new points** is selected, the user is prompted to select individual pixels. This option works well when choosing pixels near the boundaries of the object. The process is terminated using a double-click of the mouse. The points will be illuminated as blue stars. Again, the **Select** button is pressed to confirm the selection.

One or more trial regions may be deleted by clicking on a rectangle or a set of points, which will then change the color from blue to red. The **Delete selected** button is used to delete the selected regions.

The next drop down menu provides the options **HSV** or **RGB**, which classifies the pixels based on either HSV color space or RGB color space, respectively. It is possible use one colorspace for one phase and a different colorspace for another phase from the same image.

The **Fill Holes** option is used to fill holes in the working phase. For conventional concrete specimens, this option is appropriate for air voids or aggregates, but not paste.

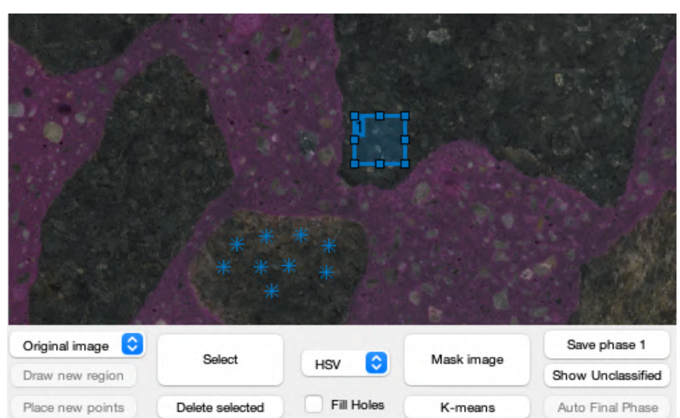


Fig. 2. User interface with menu options and two trial regions shown in blue.

The **Mask image** button runs the color thresholding algorithm and provides a masked image of all foreground pixels and an inverse mask of all background pixels. The area fraction of the foreground pixels is computed and displayed in the Command Window. New regions or points may be added using the background image or the original image.

The **K-means** button allows the user the try the automatic k-means clustering algorithm to segment the image. The user-defined trial regions are not used as input to run this algorithm.

The **Save phase** button saves the phase and allows the user to start the process again with the next phase.

The **Show unclassified** button masks all the pixels that have been classified during the session and illuminates the pixels that have not been classified as any phase. This option can be displayed at any time during the session.

The **Auto final phase** button (optional), automatically classifies all unclassified pixels as the last phase.

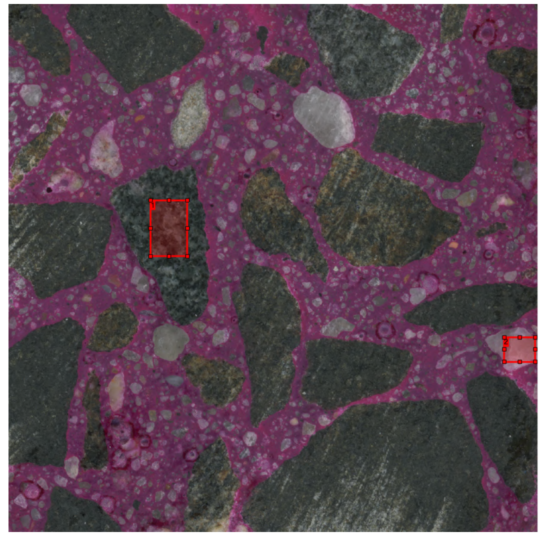
When the Auto final phase button is selected, or if the user manually classifies the last phase of the image, an image of each phase is displayed.

III. Results

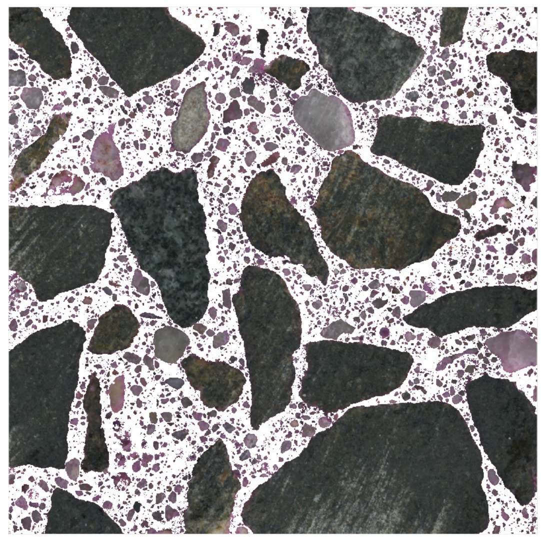
Figure 3 illustrates the results of a concrete sample using two-phase segmentation of aggregates and paste. The sample surface was stained with a 1% phenolphthalein solution as described in Section II-C. Figure 3(a) shows the original image as well as two trial rectangular regions of aggregates. The HSV and hole filling options were selected. Figure 3(b) and (c) shows the resulting aggregate and paste phases using only the two rectangular regions shown in Figure 3(a). After zooming in the image to closely inspect the edges of the aggregates as well as the fine aggregate, it appeared that the vast majority of pixels belonging to the aggregates phase were correctly classified.

The results were then compared using the automatic k-means clustering algorithm. Figure 4(a) shows a zoomed-in subset of the image in Figure 3(c) while Figure 4(b) compares the same region using the k-means algorithm. It is clear from the image that the k-means algorithm did not correctly classify much of the fine aggregates or some of the edges of the course aggregates. The same results were found throughout the sample.

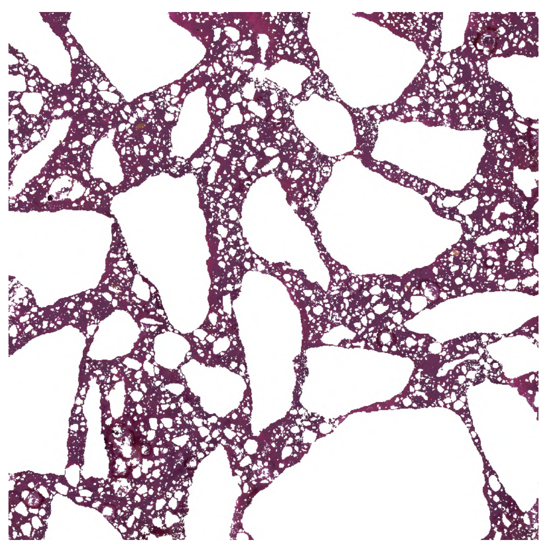
Figure 5 shows an example of results from a sample that was analyzed as a two-phase system of air voids and concrete. The target air entrainment for this mixture was 6%. The air voids were illuminated using orange chalk. The user selected three regions: a rectangular region from the largest air void (shown as a blue rectangle), and a series of points from smaller air voids from two other regions, shown as clusters of blue asterisks. The RGB mask algorithm computed 6.2% air voids while the HSV mask algorithm computed 6.0% air voids. Both the RGB and HSV masks looked very similar. In this example, the air voids were the phase of interest while the cement paste and aggregates together were classified as the background.



(a)

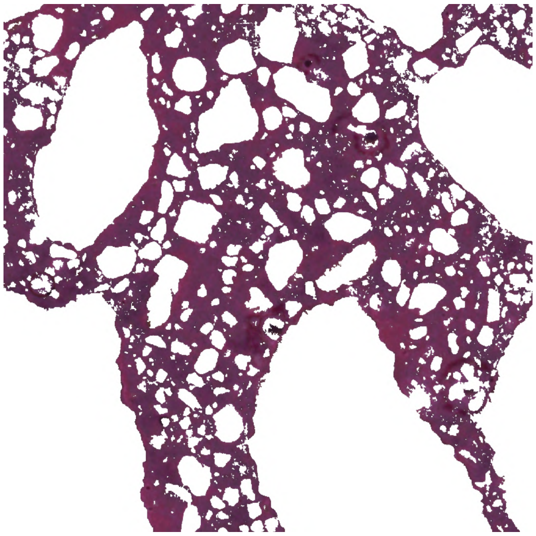


(b)

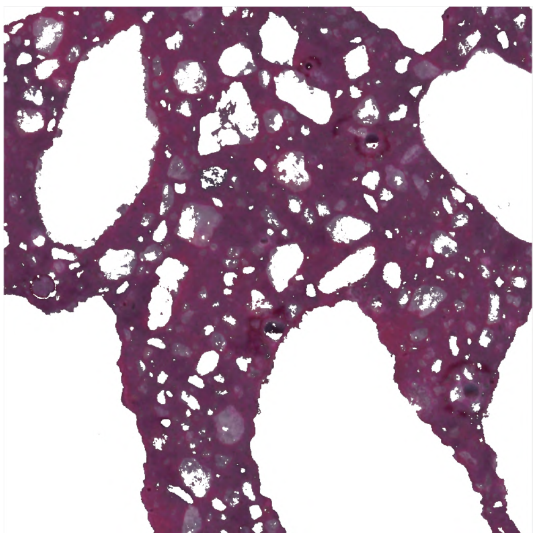


(c)

Fig. 3. Example segmentation of a two-phase (aggregate/paste) concrete sample: (a) original image with two trial regions of aggregate indicated by red rectangles, (b) result of aggregates phase, and (c) masked image showing the paste phase. Image width = 1.73 in.



(a)



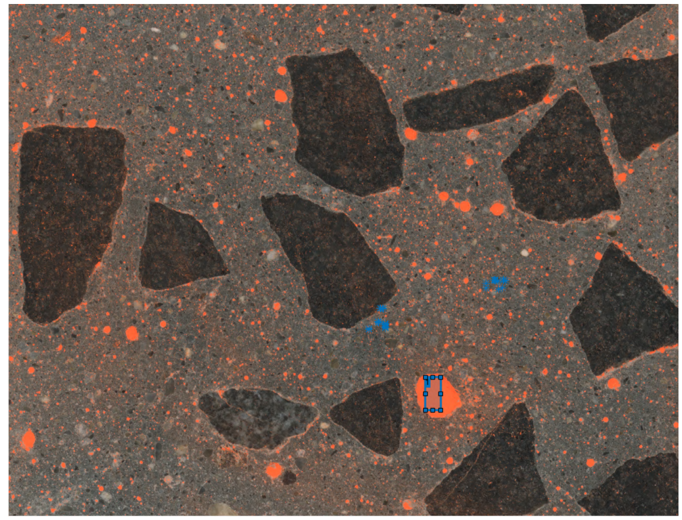
(b)

Fig. 4. Zoomed in subset of the sample in Figure 3 showing the paste phase using (a) the proposed algorithm and (b) the automatic k-means clustering algorithm. As seen in the image, when using the k-means algorithm, nearly half of the area of the fine aggregate was incorrectly classified with the cement paste. Image width = 0.55 in.

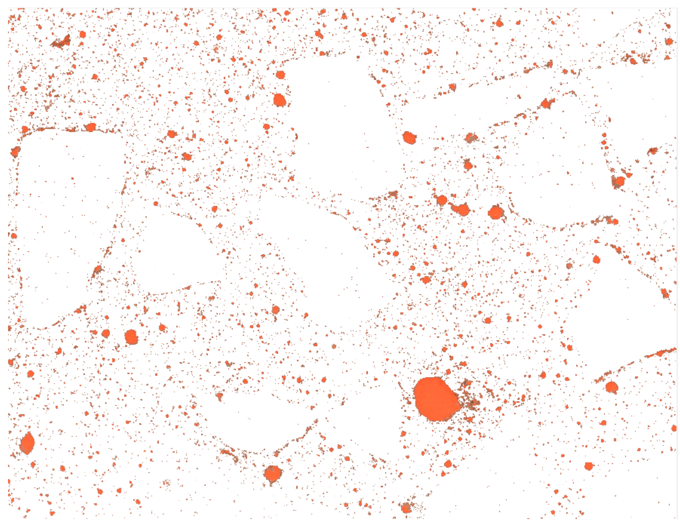
IV. Discussion

One of our goals was to extend our previous work, which used grayscale image thresholding methods to quantify the air void system in concrete [14]. Samples for this prior work were prepared by treating the surface with a black acrylic ink followed by pressing a white powder into the voids. However, using the proposed color-based algorithm, the sample preparation for air void analysis is far easier since the sample surface does not need to be stained or painted. Plenty of color distinction is provided by pressing colored chalk into the air voids without treating the remaining concrete surface.

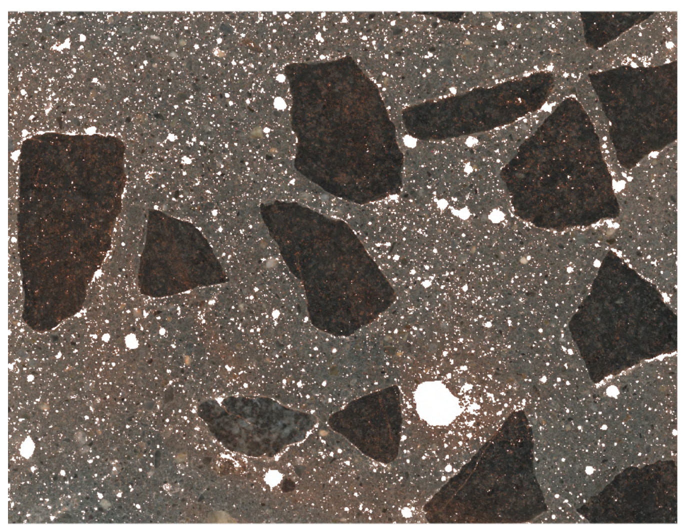
When segmenting aggregates, especially fine and multi-colored aggregates, we used using a 1% phenolphthalein solu-



(a)



(b)



(c)

Fig. 5. Example segmentation of a two-phase polished concrete sample with a 1-2 μm neon orange chalk pressed into the air voids: (a) original image showing three trial regions indicated by a rectangle and two clusters of point selections, (b) result of air void phase, (c) masked image of background showing aggregates and paste. Image width = 1.9 in.

tion to stain the cement paste to provide color distinction. At times, the aggregate surfaces were contaminated with the phenolphthalein solution, especially lighter and fine aggregates. Despite the occurrence of this contamination of pink/purple color, our proposed application was able to distinguish between aggregates and cement, as shown in Figure 4(a).

When possible, we recommend capturing sample images using a flatbed scanner, which, given the resolution of the scan, makes it possible to easily convert the area fraction of each phase to a dimensional area (e.g. mm^2 or in^2). As an alternative, if an image is captured using a camera, a reference dimension should be provided with the image in order to scale the areas accordingly. If only fractional areas are needed, then a reference dimension is not needed.

Our color-based application is user-driven, i.e., not automatic. However, we found that when determining the area fraction of a particular phase, the results tended to converge no matter which trial regions were selected. If the results did not converge, the issue usually stemmed from poor sample preparation. Either way, the masked images provided clear confirmation of successful or failed results. The k-means clustering algorithm, which is automatic, was provided with the application so that the user can make a direct comparison. However, we found that the k-means algorithm was more sensitive to imperfections in sample preparation, as seen in Figure 4(b). For example, the k-means algorithm missed the aggregates that were contaminated with the pink phenolphthalein stain, but the proposed color-based algorithm did not miss these pinkish stained aggregates. Furthermore, the k-means algorithm incorrectly classified the gray and black squares from the test sample (described in Section II-B) as a single phase while also splitting the red squares into two different phases composed of light red and dark red pixels. At times, the k-means algorithm provided good results, but there is no way for a user to provide input for the purpose of eliminating errors.

In general, only a few trial regions are needed to successfully segment a phase. For example, the segmentation shown in Figure 5 required the rectangular selection of one large air void and a few more point selections of smaller air voids. For the best results, we recommend that pixels close to the boundaries of other phases are selected as part of the trial regions.

The application allows the user to work interchangeably with RGB and HSV colorspaces. It is possible to switch colorspaces for different phases in one session. For most of the images of concrete, it appeared that using HSV colorspace led to more accurate results. However, the masking feature typically makes it obvious to the user when it is appropriate to try a different colorspace. For example, the six-phase test sample described in Section II-B required that the RGB colorspace be used to segment red squares. If HSV colorspace was used to segment the red squares, several blue pixels would erroneously be included in the classification. For the remaining colors, results using RGB and HSV colorspace were nearly identical. For a variety of samples, some trial and error may be needed, but because foreground and background images are created, it is visually clear to the user if a difference colorspace or different trial region should be used.

Recently, Matlab released a built-in app that performs color-based thresholding by allowing a user to define trial regions in a manner somewhat similar to that presented in this paper [17]. The Matlab app is interactive and enjoyable to use, but it has some limitations. It does not allow the user to define the number of phases in the image. Rather, it operates with two-phases only, making the analysis cumbersome for multi-phase materials. Hole filling is not an option, but we have found that when segmenting aggregates or air voids, hole filing is a time-saving feature as it tends to reduce the number of trial regions needed to classify a phase. The Matlab app is more graphically intensive. On a modern laptop, we found that it stalled when using the 4800 dpi images. Our proposed application is designed for multi-phase materials, allows hole filling, and runs quickly even when using large file sizes common in high-resolution images.

Since users of the proposed Matlab application provide information about how a human processes color and objects in an image, the app could be set up for further modification and AI training. Students can open and modify the code and potentially build an understanding with coding, AI, and machine learning. This app can be incorporated into engineering courses as well as provide a practical and accessible means to perform structural forensics and materials analysis.

V. Conclusion

A color-based algorithm and user-friendly application was developed in Matlab to perform color-based segmentation of the phases of concrete. While this application was developed for images of concrete, it can be used on a variety of multi-phase materials, provided that each phase can be distinguished by color. Upon providing the number of phases present in an image, the user is prompted to select one or more trial regions that belong to a single phase. Depending on user selection, the program determines the RGB or HSV color values of the trial pixels and segments the image accordingly, using a mask to highlight foreground and background pixels. This process allows the user to visually confirm the results before moving on to the next phase. Each phase of the material as well as its area fraction is saved and displayed. A package of Matlab functions are provided as supplementary material that may be opened and modified, which provides an accessible tool for engineering education.

References

[1] ASTM Standard c457/c457m-16. Standard test method for microscopical determination of parameters of the air-void system in hardened concrete, doi: 10.1520/c0457_c0457m-16. Available at https://www.astm.org/(2022/08/11)

[2] ASTM Standard c856/c856m-20. Standard practice for petrographic examination of hardened concrete, doi: 10.1520/c0856_c0856m-20. Available at https://www.astm.org/ (2022/08/11)

[3] K.W. Peterson, Automated air-void system characterization of hardened concrete: helping computers to count air-voids like people count air-voids - methods for flatbed scanner calibration. Ph.D. thesis, Michigan Technological University (2008)

[4] D. Zalocha, J. Kasperkiewicz, Cement and Concrete Research **35**(10), 2041 (2005)
[5] M. Radlinski, J. Olek, K.P. Q. Zhang, Concrete p. 64 (2010)
[6] K.W. Peterson, R.A. Swartz, L.L. Sutter, T.J. Van Dam, in *Transportation Research Record*, vol. 1775 (2001), vol. 1775, pp. 36–43
[7] L. Biehl, D. Landgrebe, Computers & Geosciences **28**(10), 1153 (2002)
[8] Multispec - a freeware multispectral image data analysis system. Available at https://engineering.purdue.edu/~biehl/MultiSpec/index.html (2022/08/03)
[9] Imagej - image processing and analysis in java. Available at https://imagej.nih.gov/ij/ (2022/08/03)
[10] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, M.S. Lew, Neurocomputing **187**, 27 (2016)
[11] A. Krizhevsky, I. Sutskever, G.E. Hinton, Advances in neural information processing systems **25** (2012)
[12] G. Letjens, Medical Image Analysis **42**, 60 (2017)
[13] Y. Song, Z. Huang, C. Shen, H. Shi, D.A. Lange, Cement and Concrete Research **135**, 106118 (2020)
[14] P. Fonseca, G. Scherer, Materials and Structures **48**(10), 3087 (2015)
[15] J.A. Hartigan, M.A. Wong, Journal of the royal statistical society. series c (applied statistics) **28**(1), 100 (1979)
[16] A.R. Smith, ACM Siggraph Computer Graphics **12**(3), 12 (1978)
[17] Segment image and create mask using color thresholder app. Available at https://www.mathworks.com/help/images/image-segmentation-using-the-color-thesholder-app.html (2022/08/03)