Development of Four-Square Fiducial Markers for Analysis of Paper Analytical Devices

Jenna Wilson[*a], *Tabitha Ricketts*[b], *Ian Bentley*[c], *Ewa Misiolek*[a]

[a] *Department of Mathematics and Computer Science, Saint Mary's College, Notre Dame, IN*
[b] *Department of English, Saint Mary's College, Notre Dame, IN*
[c] *Department of Chemistry and Physics, Saint Mary's College, Notre Dame, IN*

*Students: *jwilso01@saintmarys.edu, tricke01@saintmarys.edu*
Mentor: ibentley@saintmarys.edu, misiolek@saintmarys.edu

ABSTRACT
Fiducial markers are used in image processing to determine locations of interest based on fixed points of reference. There are a number of applications for these markers across various fields ranging from advertising to radiation therapy. The four-square fiducial markers discussed in this manuscript allow for the determination of locations of interest on digital images. These new markers are easily detected, provide information about image orientation, and allow for local color sampling. The markers are intended for use in a pharmaceutical assessment process in which images of colorimetric chemical tests are taken by a smart-phone in the field, uploaded to a database, and analyzed to collect quantitative information about the colors resulting from the tests.

KEYWORDS
Image Processing; Digital Imaging; Fiducial Markers; Image Thresholding; Color Calibration; MATLAB; Paper Analytical Devices

INTRODUCTION

In 2003, the World Health Organization (WHO) estimated that up to 25% of all medications consumed in developing countries were either counterfeit or substandard.[1] While the quality of pharmaceuticals is typically validated in quality control laboratories run by manufacturers and government agencies including the WHO,[2] the validation process is often expensive, time-consuming, and generally inaccessible to the everyday consumer, especially in developing countries. To combat this issue, Paper Analytical Devices (PADs) provide an efficient, low-cost test to identify active ingredients in such medications, see *e.g.* Weaver *et al.*[3,4] The PADs Project is an interdisciplinary research effort between Saint Mary's College and the University of Notre Dame that focuses on developing PADs to test for specific ingredients indicative of substandard pharmaceuticals.[5] Each PAD is approximately the size of a business card and contains a series of cellulose paper strips, called *lanes*. The lanes are pretreated with chemical reagents that, upon combination with certain ingredients common to a given pharmaceutical, yield colorimetric products as shown in **Figure 1**.

To test a suspect medication with a PAD, a user first applies the medication, typically as a powder or crushed pill, along a specified track that runs through the middle of the lanes, ensuring that a sufficient amount of medication is applied to each lane. Next, the user dips the PAD into water such that the blue area at the bottom of the PAD is submerged and holds it there until a blue product forms at the top of the rightmost lane, which indicates that water has travelled to the top of the lanes. When the bottoms of the lanes are immersed, the water travels up the cellulose lanes via capillary action and combines the reagents and medication, resulting in a measurable colorimetric reaction.

Each lane tests for a specific ingredient. The resulting color in each lane indicates whether the corresponding ingredient is present in the test medication. At present, there are multiple variations of the PAD, including versions that test for common ingredients in specific medications, including antibiotics and antimalarials, as well as a "general" PAD that tests for generic ingredients, such as inappropriate fillers known to be common in counterfeit and substandard medications. Depending on the version of PAD, each lane might test for a different ingredient or multiple lanes might test for the same ingredient. Repeated tests on separate lanes is often used to verify results in the case that medication is not applied evenly across each lane; for example, if more medication is applied to the first few lanes than the last few.

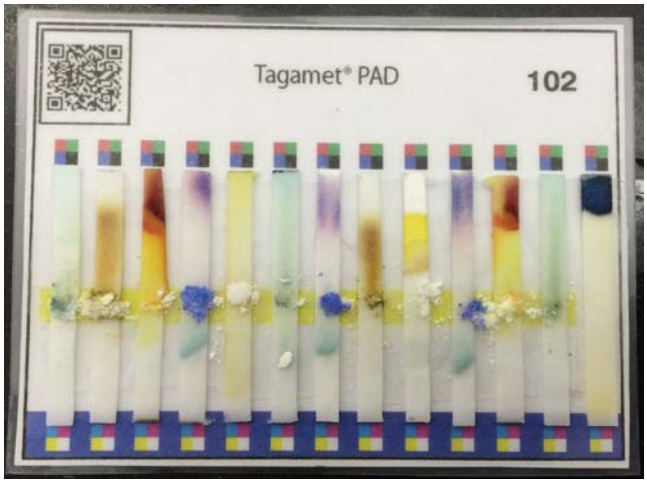


Figure 1. Smart-phone generated image of a completed PAD with 13 lanes, produced by E. Barstis. The colorimetric products resulting from chemical reactions between the pretreated reagents and pharmaceutical ingredients occur at various positions along the lane.

Each test might also produce a different type of colorimetric result. Some tests might turn one color when a certain ingredient is present and turn a different color when the ingredient is not detected. Others might turn a single color that varies in intensity in response to different concentrations of a target ingredient. For example, one test might turn the lane black in the presence of an ingredient and turn green when the ingredient is not detected while another might transition from yellow to dark orange when high concentrations of the ingredient are present. This can make it difficult for an untrained user to interpret the results of the PAD, especially for tests where the difference in color is very slight.

Image analysis software has the potential to introduce major improvements over the current use of manual PAD analysis, as it fully circumvents the optical subjectivity of a user. We are developing computational image analysis programs in MATLAB[6] to objectively attain quantitative colorimetric data and allow for rapid analysis of a large number of images in order to more quickly and more accurately assess the results of reactions on a PAD. Ultimately, the goal is to allow an untrained user to conduct a determinative analysis of the colorimetric results of a PAD using a common smart phone.

In the planned system, the user will simply take a photograph of the PAD that has been used to test a medication and has produced colorimetric results, referred to as a "completed" PAD and shown in **Figure 1**, and upload it to our database via text message. Once the image is received, the software will first identify the location of the lanes, and then analyze the results of the PAD by comparing regions of interest with cataloged color data for known reactions. Most tests are set up such that the colorimetric reaction occurs in the area of the lane above the applied medication, but because some tests result in a reaction immediately where the medication is applied, examined regions of interest include the areas above and including the applied medication, separately. Finally, the program will determine what ingredients have been detected and notify the user of the results in less than one minute.

The primary difficulty in developing such software lies in the initial step of identifying the location of the lanes on the PAD, especially when standard image and lighting quality cannot be assumed. We have solved this problem and successfully automated lane identification by developing specialized fiducial markers that each contain four colors and placing them at the top and bottom of each lane, as shown in **Figure 1**. This manuscript addresses the development of these fiducial markers and their application to not only lane identification, but also image orientation and lighting-specific color calibration.

The following sections of this paper discuss the fundamentals of digital imaging relevant to our application, the use of fiducial markers in image processing, and the development of the new fiducial markers used on the PAD. The merits of the four-square markers will also be discussed, along with the results from a recent field-test. **Appendix A** contains an algorithm for finding four-square fiducial markers. **Appendix B** contains another algorithm for an interpolation routine used to both identify missing markers and also exclude extraneous pixels falsely identified as belonging to markers.

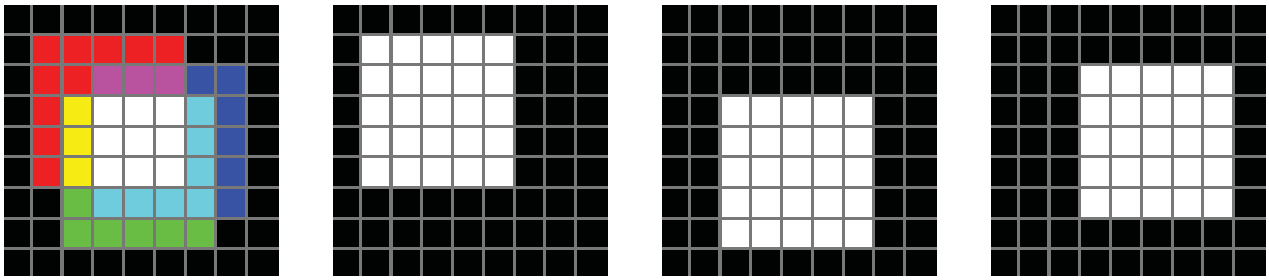
Digital imaging

A digital image is a discrete, finite representation of a continuous scene that contains both spatial and color data stored within a 2-D matrix.[7] Each element of the matrix is referred to as a *picture element*, more commonly abbreviated to *pixel*. The value stored in each pixel corresponds to the intensity of monochromatic light at that location in the image, given by the row and column of the pixel in the matrix.[8] Thus, monochromatic images such as *grayscale* images, in which color is represented as shades of gray, can be stored as a single 2-D matrix because one value is sufficient to represent the intensity at each pixel.

Color images, on the other hand, require more information. Digital color is produced visually through the superimposition of three monochromatic images and numerically through a linear combination of values from three 2-D matrices,[7] or "channels," corresponding to each of the three additive primary colors: red, green, and blue (RGB). These colors are referred to as *additive* because they create color by adding specified amounts of these primary colors to an absence of color, represented by the color black (K). In contrast, cyan, magenta, and yellow (CMY) are referred to as *subtractive* because they are created by completely subtracting RGB, respectively, from a complete presence of color, represented by the color white (W). The CMYK color space is most commonly used in printing by adding specified amounts of the subtractive color to produce color on a white medium, rather than trying to subtract amounts of the additive primaries. Likewise, the RGB color space is used primarily for digital displays by adding RGB light at varied intensities to an otherwise black screen, as is represented when no display is turned on. While other color spaces exist for digital images, the work discussed in this manuscript uses RGB images with channels containing values spanning from 0 to 255 and focuses on the eight "pure" colors, namely RGBKCMYW, created by the combinations of RGB values provided in **Table 1** and illustrated in **Figure 2**. The use of these eight colors will be discussed in a later section.

	Red Value	Green Value	Blue Value
Black	0	0	0
Red	255	0	0
Green	0	255	0
Blue	0	0	255
Cyan	0	255	255
Magenta	255	0	255
Yellow	255	255	0
White	255	255	255

Table 1. RGB values corresponding to each of the eight "pure" colors in the RGB color space.



(a) RGB color image **(b)** Red pixels of image **(c)** Green pixels of image **(d)** Blue pixels of image

Figure 2. Demonstration of a 9-pixel-by-9-pixel composite image and the corresponding RGB channels.

As the range of data able to be stored, and thus the range of potential colors displayed in a digital image, depends on the file format, it was necessary to consider image file formats that would best preserve image color and clarity. For various reasons, our programs have been calibrated to and tested with Joint Photographic Experts Group (JPEG) images. The compression algorithm used to store JPEG images was originally created to generate photo-realistic images,[9] and thus minimizes data loss when storing an image.

Moreover, JPEG images generally use 24-bit color, meaning that they have the capacity to store values for and generate $2^{24} = 256^3 = 16{,}777{,}216$ colors. Thus, JPEG images can store and display all combinations of values, from 0 to 255, in the RGB color space without incurring losses due to color correction. Although alternate formats can be used to store 24-bit data and higher, the mainstream use of JPEG images by smart phones and other digital photography devices, in addition to the qualities previously mentioned, were preferable for the development and implementation of our fiducial markers.

Object identification and fiducial markers

As previously discussed, the primary problem in developing software to analyze images of PADs was automating lane identification. There are two approaches for identifying objects or locating regions of interest in an image that were considered. One makes use of pre-installed fiducial markers (see *e.g.* Kato[10] and Naimark[11]) while the other, called *point* or *feature matching*, involves the analysis of various object components and compares those to components found on a reference image (see *e.g.* Davison[12] and Stricker[13]). This section discusses both approaches, including the factors that led to the use of fiducial markers on the PADs, and demonstrates some current, more common uses of fiducial markers in object identification.

Feature matching is a fast technique for identifying object components in an image given a reference image despite clutter, scale change, or plane rotation. MATLAB's Computer Vision ToolBox provides functions, such as `detectSURFFeatures` and `matchFeatures`, to implement this technique with efficient feature detection and comparison algorithms.[6]

Feature matching was deemed ill-suited for this project for two main reasons. The most obvious problem with this technique is that the MATLAB implementation requires that both the sample image and reference image are grayscale images. As the overall goal of this project is to quantify color produced by chemical reactions, it did not seem logical or beneficial to use grayscale images.

Possible workarounds were considered but not implemented as they all posed more problems than solutions. Additionally, a reference image would likely have to come from the user in order to accurately detect and compare the features of the PAD, due to differences in image quality that occur simply from environmental differences in the camera and lighting. Not only would this provide additional work for the user, but it would also make for a less reliable program, as it is impossible to assure that every user will send a sufficient reference image. On account of these difficulties, an approach utilizing fiducial markers is preferable for identifying lanes on images of PADs.

The fiducial markers indicate a fixed point of reference or measure in image processing. Fiducial markers can be implanted to add precision for localized radiation therapy,[14] included in printed circuit board design to provide precision when mounting boards with automated equipment,[15] or used in computer-based object recognition and tracking. Bar-codes and QR codes are commonly used fiducial markers.

(a) Examples of bar-codes.[16] **(b)** Examples of QR Codes.[17,18]

Figure 3. Commonly seen fiducial markers used for object identification and data representation in business and advertising.

The term *bar-code* typically refers to a series of vertical lines of varying thickness and spacing, as seen in **Figure 3a**. These are used primarily for product identification and data representation. This type of bar-code provides a one-dimensional representation of data, while QR codes, as seen in **Figure 3b**, provide a two-dimensional representation of data. A standard QR code contains three large corner markers that are used to indicate position and orientation. These markers are composed of three concentric squares in

black, white, and black. A smaller marker of the same type is often included near the bottom-right of a QR code and sometimes additional concentric square markers are included to allow for alignment. These fiducial markers and similar variations were considered for use on the PAD, but proved difficult to identify in our application. The shortcomings of these markers will be discussed in detail in the next section.

METHODS AND PROCEDURES

The principle motivation for using fiducial markers is to identify lane locations, but it proved nearly impossible to identify any lanes or markers consistently in images with poor orientation, such as skew, rotation, or excess background, or those taken in extremely dim or varied lighting. As such, the desired applications and functionality of the markers grew to encompass solutions to these problems. Moreover, since determining a lane's location requires identifying the markers both above and below it, lanes will go unidentified unless the markers are consistently identified. This section includes a discussion of these problems, the process of developing a fiducial marker to solve them, and the ways in which they are solved in the implementation of our final fiducial marker design.

Before attempting to identify lanes, an image must be oriented correctly in order to pair the markers appropriately according to lane, but this is impossible to do without some additional information or reference point to orient the image. We first considered adding a unique marker to indicate the top-left of the image, but instead decided to use two sets of markers to designate the top and bottom of the PAD.

There are three steps used to prepare an image for lane identification: major rotations of 90°, 180°, or 270° are used to account for images taken or read sideways or upside down; cropping is used to eliminate excess background; and minor rotations by less than 90° are used to adjust for skew. The image of a PAD is cropped based on the locations of the outermost top and bottom markers. Similarly, the degree of the minor rotations can be calculated and accounted for based on the slope between the outermost top or bottom markers. However, since this technique requires the position of the outermost markers to be known, the program must first search for and find the markers before orienting the image and identifying the lanes themselves.

Although this is a simple workaround to implement, markers are rarely found in images with poor color quality. Color calibration can be used to correct for imperfect color due to the quality of printing, image capturing, and other environmental variables, as well as more drastic color imperfections, such as in images that lack sufficient brightness, contrast, or both. While global color adjustment can be used to modify the color of the entire image in order to aid marker identification, it does not sufficiently adjust the color in images containing vast light gradation. The effects of lighting variability is likely to be the most severe when a picture is taken with the flash on, resulting in bright central lanes and dark outer lanes. Additionally, if a portion of a light source is obstructed, then shadow can be cast on the image. By sampling colors on the fiducial markers, the colors of each lane can be calibrated individually, increasing contrast and allowing for more accurate data.

The initial fiducial markers designs developed and tested to address these concerns were composed of three concentric squares, similar to those seen in the corners of QR codes. These squares were edited to incorporate various patterns of RGB and CMY colors, such as those seen it **Figure 4**, to designate the top and bottom markers, respectively, and to allow for local color sampling.

(a) Fiducial marker used in QR codes **(b)** RGB fiducial marker **(c)** YMC fiducial marker

Figure 4. Fiducial marker design composed of concentric squares, including that used in QR codes and variations tested on the PADs.

Concentric square markers of this type are identified by a specific sequence of color transitions among adjacent pixels. That is, the algorithm used to identify these markers would first identify continuous regions of the colors used in the markers and then group these regions and label them as markers if they were bordering each other in the appropriate sequence. Unfortunately, the adjacent colors on the concentric square markers had substantially blurred edges caused by both printing and photographing. This resulted in muddled colors, making it difficult for the program to consistently find the markers since the change in color affected the program's ability to identify continuous regions of any color or bordering sequences of colored regions.

Eventually, two new fiducial markers were developed consisting of four adjacent colored squares, rather than concentric squares. The four-square marker composed of adjacent RGBK squares, shown in **Figure 5a**, is used to identify the top of each lane while that composed of CMYW squares, shown in **Figure 5b**, corresponds to the bottom of each lane. These colors have been chosen in order to contrast the background on the PAD. These four-square markers are easily, efficiently, and accurately identified through regional comparisons of four corners about a central pixel, especially when compared to the previous method used for the concentric markers.

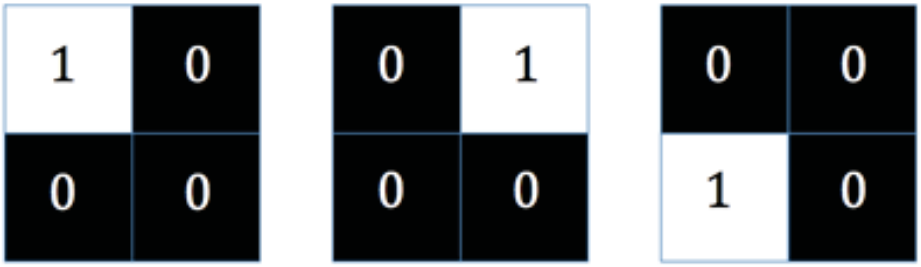
(a) RGBK fiducial marker **(b)** CMYW fiducial marker

Figure 5. Four-square fiducial marker design composed of adjacent colored squares.

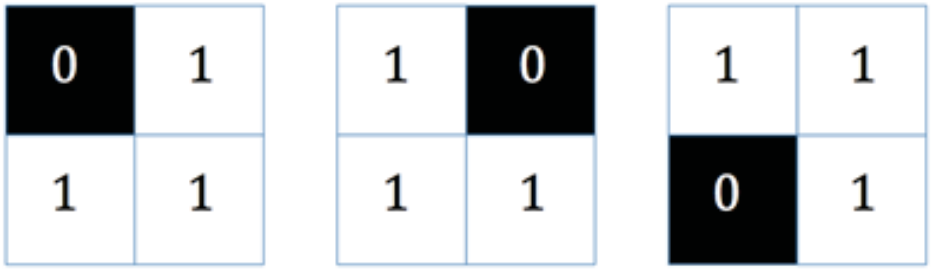
The identification of regional comparisons can be demonstrated through a binary representation of each RGB channel of the markers. The binary representation is attained by assigning logical values of 1 (true) to pixels with the maximum value of 255 in that channel's color and 0 (false) to all others. This representation generates a 3-bit binary image, comprised of the superposition of binary (1-bit) matrices in each of the RGB channels, resulting in the $2^3 = 8$ pure colors. Both the binary (1-bit) and RGB (24-bit) values are provided in **Table 2** and each channel of the binary representation of the four-square marker is illustrated in **Figure 6**.

Color	RGB values	Binary values
Black	[0 0 0]	[0 0 0]
Red	[255 0 0]	[1 0 0]
Green	[0 255 0]	[0 1 0]
Blue	[0 0 255]	[0 0 1]
Cyan	[0 255 255]	[0 1 1]
Magenta	[255 0 255]	[1 0 1]
Yellow	[255 255 0]	[1 1 0]
White	[255 255 255]	[1 1 1]

Table 2. RGB and binary values corresponding to each of the eight "pure" colors in the RGB color space.



(a) Binary matrices of each RGB channel for RGBK fiducial marker.



(b) Logical matrices of each RGB channel for CMYW fiducial marker.

Figure 6. Binary matrix corresponding to the red (left), green (middle), and blue (right) channels of the four-square fiducial markers.

The left-most combination of squares in **Figure 6a** shows the 2-D binary matrix of the red pixels of the RGBK marker. Note that the only matrix element containing a value of 1 corresponds to the red quadrant of the RGBK marker while all other quadrants return 0s. Similarly, the left-most square in **Figure 6b** shows the binary matrix of the red pixels of the CMYW marker. As discussed in the introduction, cyan is created in the absence of red, hence, the quadrant corresponding to the cyan quadrant of the CMYW marker returns a 0 in the red channel while all other quadrants contain values of 1.

In order to generate the binary matrices of a captured colored image, we utilize image thresholding to convert each of the RGB values in the pixel to a binary value by assigning 1 to color values of a pixel exceeding the threshold and 0 otherwise. That is:

$$L(x,y,c) = \begin{cases} 1 & \text{if } I(x,y,c) > T \\ 0 & \text{otherwise} \end{cases} \qquad \textbf{Equation 1.}$$

where the color image, $I(x,y,c)$, contains RGB values at each pixel and the corresponding binary image, $L(x,y,c)$, is determined using a threshold value, T.[19] Here, x denotes the horizontal position and y the vertical positions measured from the top left, while c designates the channel, either red, green or blue. For example, a pure red pixel located in the top-left corner of an image would correspond to $I(1,1,[255,0,0])$. Furthermore, the red value of this pixel would correspond to $I(1,1,1) = 255$. In this manner, each of the three color matrices are converted into binary matrices using one threshold value.

Thresholding allows a program to recognize "impure" colors that occur whenever a fiducial marker does not contain the expected RGB values. Note that if each pixel's RGB values are at either of the appropriate extremes, that is, the value of each channel is either 0 or 255, then any threshold values ranging from 1 to 254 will result in the same 2-D binary matrices shown in **Figure 6**. Further, the odds of incorrectly identifying random noise as a specific orientation of one of the two four-square color markers is low, specifically 1 in 8^4 = 4096, since each of the four quadrants must be one of eight colors, and the colors of each quadrant are determined independently. When considering both colored four-square markers, these odds remain low, but change to 2 in 8^4 = 4096, or 1 in 2048. As mentioned earlier, the program searches for markers before adjusting orientation, raising the odds of identifying random noise as markers to 8 in 8^4 = 4096, or 1 in 8^3 = 512, to account for both markers in any of the four rotations; however, it will be discussed later how the program accounts for noise identified as incorrectly oriented markers and thus eliminates

the need to consider all four rotations in these chances. This is important because the photographic images taken in the field will vary as a result of the quality of printing on the PAD and the digital image taken, so a low chance of identifying stray pixels as markers allows images to be analyzed accurately with less dependence on the quality of the image.

Our program searches for markers using a range of threshold values to accommodate for environmental variations. These values are not predetermined, but are set according to the variations of color within each specific image. At each threshold, the program iterates through the pixels of the image, at each comparing the RGB values of the pixels a specified distance to the upper-left, upper-right, lower-left, and lower-right. If the corresponding binary image indicates a sequence of RGBK or CMYW markers, then the position of the central pixel is saved as a potential marker and records RGBK and CMYW sequences that correspond to major rotations. All saved pixels are later grouped and labeled. The complete algorithm used to identify the fiducial markers on an image is included in **Appendix A**.

Once a sufficient number of markers are found to proceed with the image analysis, the image is rotated according to the color sequence that the majority of possible markers indicates. For example, if an image is rotated by 180°, the bottom-right quadrant of each RGBK marker and CMYW marker will be red and cyan rather than black and white, respectively. Thus, if the majority of possible markers were found with this color sequence, the image will be rotated accordingly. Except in extremely poor quality images, the majority of possible markers found will be pixels from the markers themselves because the probability of incorrectly identifying random noise as one of these markers is low, as previously discussed, and thus signal the necessary degree of rotation and eliminate any pixels identified with a different color sequence from the pool of possible markers. In this way, the development of these fiducial markers has resolved the problem of motivating the proper rotation without requiring an additional marker to indicate the orientation.

Figure 7 demonstrates an example the visual output of the marker identification programs run on the image in **Figure 1**. Notice that the image in **Figure 7** has been slightly rotated, cropped, and color adjusted, and indicates expected marker position and color. The expected center of each marker has been determined using the algorithm discussed in **Appendix B**. Once these adjustments have been made, the borders of the lanes can be calculated based on the marker positions and the color analysis can proceed.

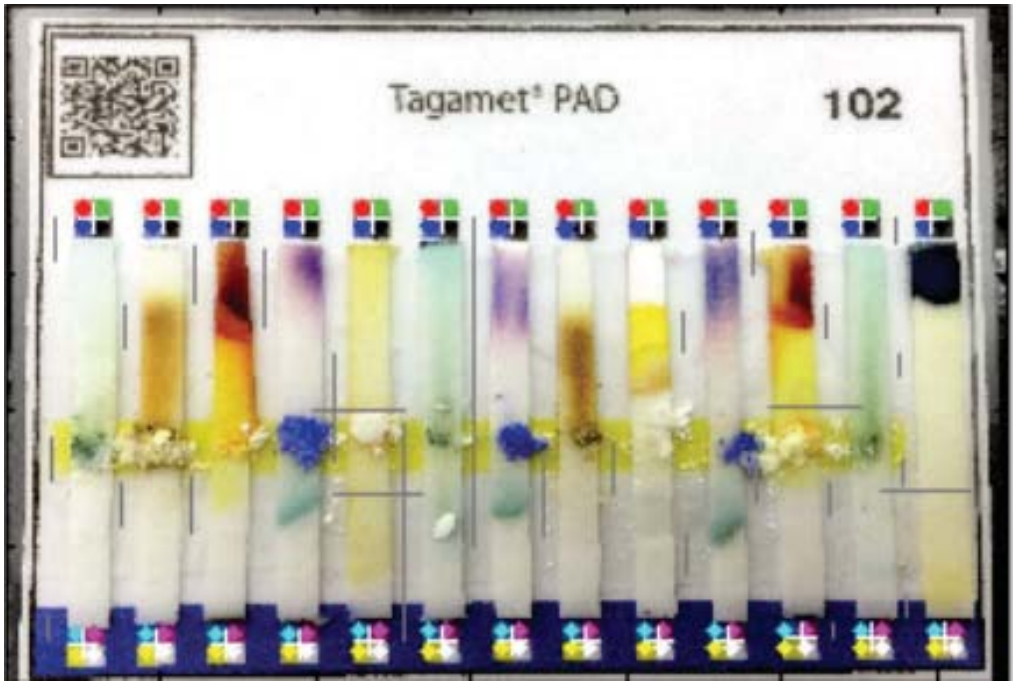


Figure 7. Analyzed version of **Figure 1**. Small colored circles have been overlaid on top of the image to indicate locations of color sampling on markers and small white cross-hairs indicate the expected center of each marker based on a linear fit.

Beyond lane identification, which was the main goal intended to be accomplished with these markers, these markers can be used to aid the color analysis through local color sampling. Under the assumption that the RGB values of the colors in the fiducial markers will provide the most extreme of values, the colors in a lane can be calibrated based on the RGB values of its markers by "stretching" the range of RGB values to contain a maximum value of 255 and a minimum of 0. For example, if the pixels in the center of the red quadrant of a lane's RGBK marker has RGB values of [220, 30, 10] and those in the cyan quadrant of the corresponding CMYW marker has RGB values of [20, 230, 225], then the red values in that lane can be adjusted so that they span from 0 to 255 rather than from 20 to 220. This can be applied to each RGB range and to each lane individually. Since color analysis of the PADs relies on a comparison of RGB values to those of catalogued data, this step standardizes the range of colors between samples and catalogued data.

RESULTS

A field-test of both the usability and computational analysis of PADs was conducted at Saint Mary's College on the 4th and 5th of March, 2015. As part of these tests, 172 iPhone images of PAD samples were taken by untrained users and sent to a database. These images were then used to test the reliability of our program. For these images, the program iterated through successive threshold values. Specifically, values of $T = 32, 64, 96, 128, 160, 192$ were used to determine the binary images and subsequently the fiducial markers. All markers were identified that contained the appropriate four-square color sequence in the quadrants at a distance of seven pixels vertically and horizontally from the center.

Of the 172 images, 80% had all 26 markers correctly identified. For an additional 6% of the iPhone images, the program was able to identify the outermost corner markers, allowing all remaining markers to be estimated using the interpolation routine described in **Appendix B**. The remaining 14% of the iPhone images had incorrectly identified lanes or no identified lanes. For 12% of the images, the corner markers were not correctly identified because all four corner markers were not present in the photograph. For the other 2%, the program did not identify any markers at all because the images were too distorted or blurry for the color sequences in the markers to be consistently detected.

DISCUSSION

Fiducial markers are needed to aid in the rapid and accurate analysis of PADs to test the validity of pharmaceuticals. Fiala[20] discusses basic criteria for measuring fiducial marker performance. For our intended use on PADs, the most critical criteria are false positive and false negative rates, vulnerability to lighting conditions, and speed performance. The markers introduced here consist of four differently colored squares placed into adjacent quadrants. The signature color combinations of markers are scanned for and if the outermost corner markers are all found, then regions of interest can be identified.

The newly developed four-square markers and the identification routine included in **Appendix A** are robust even in variable lighting situations when multiple thresholds are used, though this use comes at the cost of performance speed. Additionally, the use of a solid color background further reduces the number of false positive marker identifications. The exclusion routine included in **Appendix B** can determine rows of adjacent markers and eliminate outliers, thus diminishing the sensitivity on false positive and false negative counts. These techniques, however, are vulnerable to occlusion of a corner marker and do not provide perspective support.

The lane identification procedure makes use of many optimized MATLAB functions resulting in a program that takes about 20 seconds to identify lanes. In the future, clearly stated user instructions will be created to improve the success rate, as inexpert photography was the greatest source of incorrect lane identification in initial field-tests by untrained users. Additionally, the calibration parameters can be optimized to improve the success rate with images from a wider variety of devices as well as those from the smart phones and digital devices already tested, and speed the process further. Future development toward the ultimate goal of analyzing images of PADs will require the cataloging of colors corresponding to known chemical reactions and the creation of a color comparison program to identify reactions and notify users of the outcome.

CONCLUSION

Overall, the four-square fiducial markers developed for analyzing images of the PADs serve as simple yet effective indicators of the orientation of a PAD in an image, the location of lanes, and the colors near each lane. Based on the success of these markers, we are one step closer to providing a means for the public to test the quality of their medications.

ACKNOWLEDGMENTS

This research was sponsored by the Marjorie Neuhoff Summer Science Research Communities Grant at Saint Mary's College. We thank E. P. Aldrich, E. Barstis, and D. Marquez for many helpful discussions, and T. L. O. Barstis and K. Haas for supervising the field-tests.

REFERENCES

1. Bagozzi, D. (2003) Substandard and counterfeit medicines, World Health Organization *http://www.who.int/mediacentre/factsheets/2003/fs275/en/*.
2. World Health Organization (2007) *Quality assurance of pharmaceuticals: a compendium of guidelines and related materials. Vol. 2, Good manufacturing practices and inspection.* 2nd ed.
3. Weaver, A., Reiser, H., Barstis, T., Benvenuti, M., Ghosh, D., Hunckler, M., Joy, B., Koenig, L., Raddell, K. and Lieberman, M. (2013) Paper Analytical Devices for Fast Field Screening of Beta Lactam Antibiotics and Antituberculosis Pharmaceuticals, *Anal Chem* 85(13), 6453-6460.
4. Weaver, A. and Lieberman, M. (2015) Paper Test Cards for Presumptive Testing of Very Low Quality Antimalarial Medications, *Am J Trop Med Hyg* 92, 2-7.
5. Paper Analytical Device Project. ©2015, University of Notre Dame. *http://padproject.nd.edu/*.
6. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. ©1994-2015. *http://www.mathworks.com/*.
7. Solomon, C. and Breckon, T. (2011) *Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab*, 1-12, 49-80, Wiley-Blackwell, John Wiley and Sons, Ltd. Publication.
8. Gonzalez, R. C. and Woods, R. E. (2008) *Digital Image Processing, Third Edition*, Pearson Education Inc.
9. Qidwai, U. and Chen, C.H. (2010) *Digital Image Processing: An Algotithmic Approach with MATLAB*, 1-28, CRC Press, Chapman and Hall.
10. Kato, H. and Billinghurst, M. (1999) Marker tracking and HMD calibration for a video-based augmented reality conferencing system, in *2nd IEEE and ACM International Workshop on Augmented Reality*, 85-94.
11. Naimark, L. and Foxlin, E. (2002) Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker, *IEEE and ACM International Symposium on Mixed and Augmented Reality*, 27-36.
12. Davison, A. J. (2003) Real-time simultaneous localisation and mapping with a single camera, *Ninth IEEE International Conference on Computer Vision*, 1403-1410, IEEE Computer Society.
13. Stricker, D. and Kettenbach, T. (2001) Real-time and markerless vision-based tracking for outdoor augmented reality applications, *IEEE and ACM International Symposium on Augmented Reality*, 189-190.
14. Erickson, B. J. and Jack, C. R. Jr. (1993) Correlation of single photon emission CT with MR image data using fiduciary markers, *AJNR* 14(3), 713.
15. Carlton Industries Corporation (2004) Fiducial Mark Design Guidelines, 2-3.
16. Code 39 and Code 93 barcodes. ©Copyright Measurement Equipment Corporation. *http://www.makebarcode.com/*.
17. QR Code, ©2015 About.com. *http://desktoppub.about.com/od/glossary/g/QR-Code.htm*.
18. QR Code Linking to *www.linkedin.com/in/jwilson23*, ©2015 QRStuff. *http://www.qrstuff.com/*
19. Marques, O. (2011) *Practical Image and Video Processing Using MATLAB*, 61-100, 125-199, 365-405, IEEE Press, John Wiley and Sons, Ltd. Publication.
20. Fiala, M. (2010) Designing Highly Reliable Fiducial Markers, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32(7), 1317-1324.

STUDENT AUTHORS

Jenna Wilson is currently an undergraduate student enrolled in a five-year dual-degree engineering program at Saint Mary's College and the University of Notre Dame in Notre Dame, IN. She will graduate with a Bachelor of Science in Mathematics and a Bachelor of Arts in Philosophy from Saint Mary's College in May 2016, as well as a Bachelor of Science in Computer Science with a concentration in media computing from the University of Notre Dame the following year in May 2017. Jenna is particularly interested in computer graphics and plans to pursue a career in 3D computer animation.

Tabitha Ricketts is currently in her fifth year of a five-year dual-degree engineering program at Saint Mary's College and the University of Notre Dame in Notre Dame, IN. She graduated with a Bachelor of Arts in English Writing from Saint Mary's College in May 2015 and will graduate with a Bachelor of Science in Computer Science with a concentration in cyber security from the University of Notre Dame in May 2016. Tabitha plans to pursue a career in digital forensic analytics.

PRESS SUMMARY

Bar-code scanners, like those seen in grocery stores, are able to distinguish between products by analyzing a sequence of vertical bars. We have developed a comparable set of markers which allow us to determine the locations of interest on digital photographs The determination of these regions was a critical first step toward our ultimate goal of providing a means of verifying the quality of pharmaceuticals by sampling for active ingredients.

APPENDIX A: ALGORITHM FOR LOCATING THE FOUR-SQUARE FIDUCIAL MARKERS

This algorithm takes in an original image, detects each four-square fiducial marker and labels them using routines in MATLAB.[6] The MATLAB functions used require the MATLAB's Image Processing Toolbox and are included in the comments on the right side of the corresponding pseudocode. In-line comments are highlighted in green.

Input: digital image I of a completed PAD ▷ see **Figure 1**

Initialization

read digital image I ▷ using `imread`
set threshold values
set distance d to the number of pixels from a central pixel to its possible marker color quadrants

Global Lighting Adjustment

set $scale$ to a small number of pixels proportional to the image size to be disregarded as noise or background
initialize color value iterator i and pixel counter sum
initialize min ▷ records minimum color value for each channel
initialize max ▷ records maximum color value for each channel
for each RGB color channel C **do**
    generate histogram $H(c)$ representing the number of pixels with color value $0 \leq c \leq 255$ ▷ using `imhist`
    ▷ find minimum value
    $i \leftarrow 0$ ▷ set i to 0
    $sum \leftarrow H(i)$ ▷ set sum = $H(i)$
    while $sum < scale$ **do**
        $sum \leftarrow sum + H(i)$ ▷ add $H(i)$ to sum
        $i \leftarrow i + 1$ ▷ increment i by 1
    end while
    $min(C) \leftarrow i$ ▷ record i as minimum color value of C
    ▷ find maximum value
    $i \leftarrow 255$ ▷ set i to 255
    $sum \leftarrow H(i)$ ▷ set sum to $H(i)$
    while $sum < scale$ **do**
        $sum \leftarrow sum + H(i)$ ▷ add $H(i)$ to sum
        $i \leftarrow i - 1$ ▷ decrement i by 1
    end while
    $max(C) \leftarrow i$ ▷ record i as maximum color value of C
    adjust the range of color values from $min(C)$ to $max(C)$ to 0 to 255 ▷ using `imadjust`
end for

Major Rotation

for each T **in** the set of threshold values **do**
    ▷ apply threshold
    initialize binary image B of the same size as I
    for each RGB color channel C **in** I **do**
        for each pixel coordinate (x, y) **in** C **do**
            if $C(x, y) > T$ **then** ▷ if the color value is greater than the threshold
                $B(x, y, C) \leftarrow 1$ ▷ see **Equation 1.**
            end if
        end for
    end for

▷ determine the necessary degree of rotation
initialize rotation counters $rot0, rot1, rot2, rot3$ to 0 ▷ for 0°, 90°, 180°, 270° rotation, respectively
initialize $quadrants$
for each pixel coordinate (x, y) **in** I **do**
▷ find the sequence of colors d pixels from (x, y) given by B according to **Table 2** and store to $quadrants$
$quadrants(x, y) \leftarrow [B(x - d, y - d)\ B(x + d, y - d)\ B(x - d, y + d)\ B(x + d, y + d)]$
switch $quadrants$ **do** ▷ see **Figure 6**
case RGBK ∨ CMYW: $rot0 \leftarrow rot0 + 1$ ▷ sequence indicates no rotation
case KRGB ∨ WCMY: $rot1 \leftarrow rot1 + 1$ ▷ sequence indicates 90° rotation
case BKRG ∨ YWCM: $rot2 \leftarrow rot2 + 1$ ▷ sequence indicates 180° rotation
case GBKR ∨ MYWC: $rot3 \leftarrow rot3 + 1$ ▷ sequence indicates 270° rotation
end switch
end for
end for
$R \leftarrow I$ rotated according to $max\{rot0, rot1, rot2, rot3\}$ ▷ using `imrotate`

Locate Markers

for each T **in** the set of threshold values **do**
▷ apply threshold ▷ see lines 26-34
initialize binary image B of the same size as R
for each RGB color channel C **in** R **do**
for each pixel coordinate (x, y) **in** C **do**
if $C(x, y) > T$ **then** ▷ if the color value is greater than the threshold
$B(x, y, C) \leftarrow 1$ ▷ see **Equation 1.**
end if
end for
end for
▷ identify oriented markers
initialize binary image M of the same size as R ▷ to store pixels identified as possibly belonging to a marker
for each pixel coordinate (x, y) **in** R **do**
▷ find the sequence of colors d pixels from (x, y) given by B according to **Table 2** and store to $quadrants$
$quadrants(x, y) \leftarrow [B(x - d, y - d)\ B(x + d, y - d)\ B(x - d, y + d)\ B(x + d, y + d)]$
if $quadrants(x, y)$ = RGBK ∨ CMYW **then** ▷ see **Figure 6**
$M(x \pm d, y \pm d) \leftarrow 1$ ▷ set the neighborhood of pixels in M to true
end if
end for
end for
label each neighborhood of true pixels in M as a marker ▷ using `bwlabel`

APPENDIX B: ALGORITHM FOR EXTRACTING FALSELY IDENTIFIED FIDUCIAL MARKERS

This algorithm is used to identify groups of fiducial markers placed in a row from any extra markers (false positives), and is used to extrapolate any internally-missed markers (false negatives) using routines in MATLAB[6] and are included in the comments on the right side of the corresponding pseudocode. In-line comments are highlighted in green.

Require: locations of the centers of identified markers

Initialization

set the expected number of markers in a row
set the number of points to be used for a linear fit of markers
set the minimum R^2 fit value
set the minimum residual tolerance

Identify Actual Markers from Locations of Possible Markers

group the markers that are in the top and bottom halves of the image
for each group of markers **do**
    ▷ find the best linear fit to estimate the location of the row of markers
    repeat
        randomly select markers from the group ▷ using `randperm`
        find the linear fit of the randomly selected markers ▷ using `fit`
        decrease the minimum R^2 value slightly ▷ allows for curvature
    until sample $R^2 >$ minimum R^2
    ▷ determine which indicated markers are most likely true positives based their relationship to the best fit line
    for each identified markers **do**
        calculate residual between marker and best fit line
        if residual $<$ tolerance **then**
            ▷ this is likely a true positive
            record the locations of the marker's center
        end if
    end for
    find the markers with minimum and maximum horizontal coordinates
    determine the locations of the outermost markers
    calculate the average horizontal spacing between markers in a row
    approximate coordinates of internal markers
end for