# Segmentation of Color Maps Using Eigenvector Line-Fitting Techniques


***Alireza Khotanzad***
***Image Processing and Analysis Laboratory, Electrical Engineering Department***
***Southern Methodist University, Dallas, Texas***

***Edd Zink***
***Productivity Systems Incorporated***
***Richardson, Texas***



## Abstract

This paper presents an automatic color segmentation algorithm for paper-based topographic maps. The algorithm uses an eigenvector line-fitting technique to overcome false colors introduced by the scanning process due to RGB misalignment. which is caused by the optical characteristics of the scanner lens and is mostly evident in regions of color change. The resulting false colors render traditional color clustering schemes ineffective. This approach based on an eigenvector line-fitting technique uses the color information of pixels in a 3 by 3 window to determine the true color of the pixel. This algorithm has been experimentally verified to be robust and accurate.


## Introduction

The problem being considered here is the automatic segmentation of paper-based United States Geological Survey (U.S.G.S.) topographic maps. All U.S.G.S. topographic maps are printed on a white paper stock using standard ink colors as defined in Table 1. A sample image, shown in Figure 1, was digitized by a flatbed scanner at a resolution of 100 dots per inch. To the naked eye, this image is crisp and contains five distinct colors: white, brown, black, blue and purple. One would expect clustering in RGB space to produce five perfect clusters. Analysis of this image, however, reveals thousands of distinct colors which are neither visible to the naked eye nor exist in the original map. These extra colors are termed false colors and are induced by the scanning process due to RGB misalignment as discussed in the next section. To demonstrate the RGB misalignment phenomenon, the pixels in a small region containing a portion of the two vertical black lines representing a roadway are plotted in RGB space in Figure 2. In the absence of RGB misalignment, one would expect to see two clusters centered about the colors white and black representing the white background and the black road with a tube-like structure connecting the clusters containing aliased shades of gray pixels. This plot, however, reveals two tube-like structures connecting the clusters. The one on the left passes through green shades and the one of the right passes through purple shades showing that RGB misalignment tends to split the aliased pixels into two smaller tube-like structures that flare thus introducing false colors.

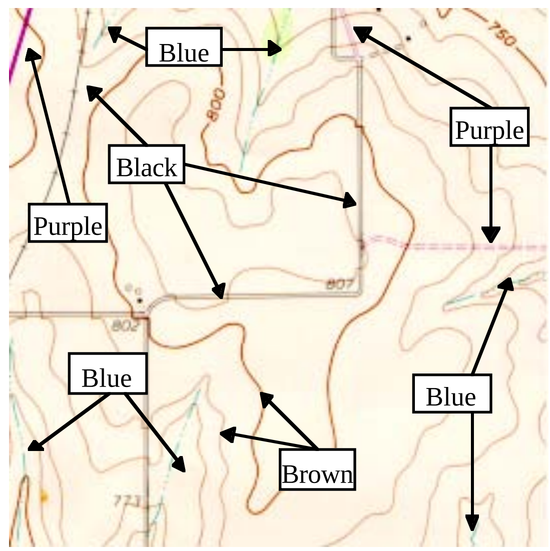


*Figure 1. Monochrome version of the color test image.*

**Table 1. Standard Inks**

| Color | Category |
|---|---|
| Brown | Contour lines |
| Black | Cultural features |
| Purple | Updated cultural features |
| Blue | Rivers |
| Green | Vegetation |

## RGB Misalignment

RGB misalignment occurs whenever the red, green, and blue color planes are not perfectly registered. Flatbed scanners digitize color images in three passes using red, green, and blue light filters. The first cause is due to a physical misalignment. If the sensor head is moved or the physical geometry between the lens, the picture and the image plane changes between color scans, the color planes will not be registered. This type of misregistration can be avoided by better scanner design and manufacturing. The second cause is due to optical misalignment and has to do with a prism effect of the lens material. As shown in Figure 3, filtered red, green, and blue light rays traveling through optical path A from the paper to be scanned and the lens are bent slightly differently by the lens thus hitting the image plane at slightly different locations (e.g., $A_b$, $A_g$, and $A_r$) with a dispersion on the order of several pixels. This effect is known as "lateral chromatic aberration" and is a direct result of the dependence of the refraction index of the lens material on the wavelength or color of the light passing through the lens. Optical misalignment is an inherent property of flatbed scanner design and is very expensive and difficult to minimize optically.

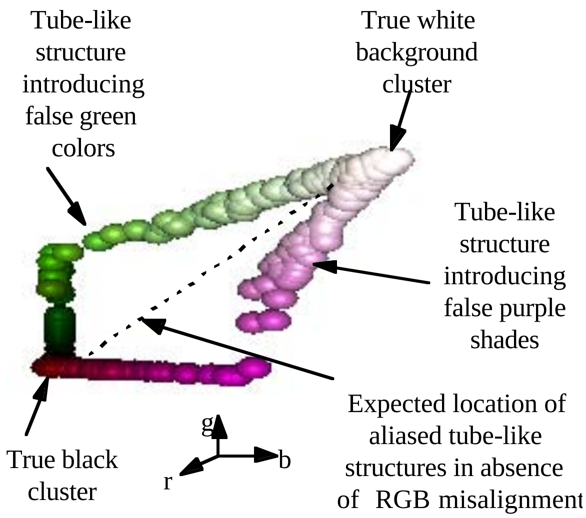


*Figure 2. Illustration of RGB misalignment induced false colors in RGB space for pixels around a vertical black line*

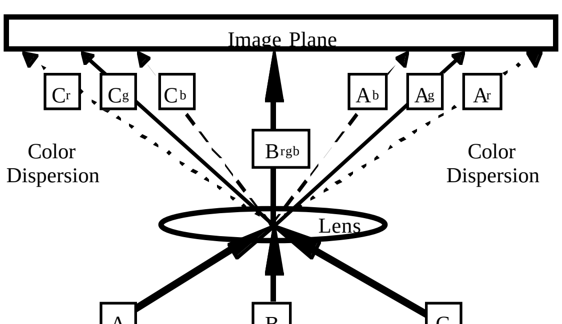


*Figure 3. Geometry of a color flatbed scanner*

RGB misalignment renders traditional color clustering algorithms ineffective. To illustrate this point, a color clustering algorithm[1] was applied to the image shown in Figure 1 which resulted in 130 clusters. Most of the clusters were false or incorrect containing pixels from different classes. Figure 4 plots a cluster that contains brown and black pixels. Other researchers have reported the effects of false colors caused by RGB misalignments. Hedley and Yan[2] noticed that "the colors in the map [image] are not distinct as would be expected." They developed a gradient thresholding scheme to overcome this misalignment that combines spatial and color space information to classify each pixel. Marcu and Abe[3] also detected RGB misalignment and plotted this phenomenon in RGB space. Their results are very similar to the RGB color space plot in Figure 2. They corrected the problem by realigning the RGB channels using intense image processing computations.

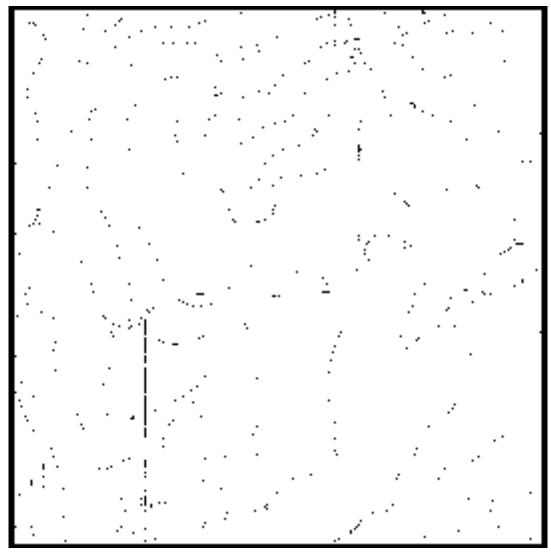

*Figure 4. RGB cluster containing brown and black aliased pixels.*

## Color Segmentation

The main emphasis of this algorithm is to correctly segment U.S.G.S. paper-based maps automatically and to be robust against any false colors introduced by the digitization process. This algorithm involves two steps. First, the image is segmented via a eigenvector line-fitting technique and the resulting clusters are smoothed using a majority rule. The final step involves clustering the resulting smoothed eigenvector image with the white background subtracted out to yield the final clusters.

The algorithm starts by centering a 3 by 3 window about each pixel in the RGB source image. This window provides a set of 9 points in RGB space. Let the set of points be denoted as:

$$\{V_i = (r_i, g_i, b_i)\}, \text{ where } i = 1, 2, \ldots, 9$$

where $r_i$, $g_i$, and $b_i$ are the red, green and blue components of $V_i$. Let $d_i$ be the perpendicular distance from the *i*th pixel to the best-fit line through the 9 points in RGB space. This is the line that minimizes the sum of the square error as shown in the following equation:

$$d^2 = \sum_{i=1}^{9} (d_i)^2$$

It can be shown that the line minimizing the above equation is parallel to the eigenvector ($\phi_r$, $\phi_g$, $\phi_b$) having the largest eigenvalue ($\lambda$) of the following scatter matrix (S)[4]:

$$S = \sum_{i=1}^{9} v_i(v_i)^t$$

Thus, the direction of the eigenvector corresponding to the largest eigenvalue (i.e. principle eigenvector) is parallel to the line that minimizes the perpendicular distance between the 9 pixels and the best-fit line in RGB space. An example is shown in Figure 5. The principle eigenvector ($\phi_r$, $\phi_g$, $\phi_b$) and eigenvalue ($\lambda$) have special meaning. First, high eigenvalues occur in regions of high color change and aliasing. Secondly, the direction of eigenvector represents the perceived color of the pixel.

The resulting principle eigenvectors are used to create an eigenvector image as shown in Figure 6. Each pixel in the RGB source image is replaced by its corresponding eigenvector that represents the best-fit line in RGB space of the pixel and its eight connected neighbors. These values are then mapped to [0, 255]. Next, a color clustering algorithm[1] is used to segment the eigenvector image resulting in slightly over 50 clusters for the considered example. These eigenvector results are shown in Figures 7 through 10 for the brown, black, blue, and purple clusters respectively. Note that the lines represented in the eigenvector segmentation results are much thicker than their corresponding lines in the RGB source image.

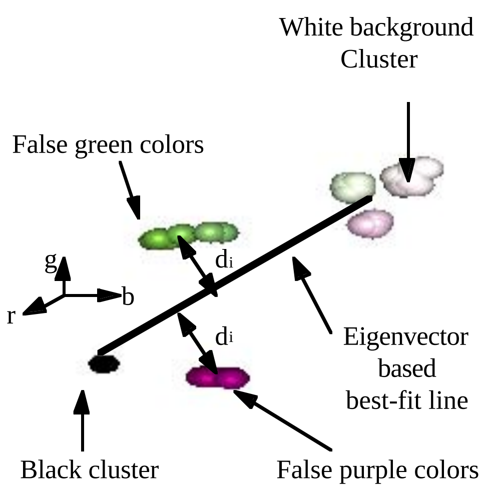


*Figure 5. Eigenvector line fitting in RGB space for a typical 3 × 3 window.*

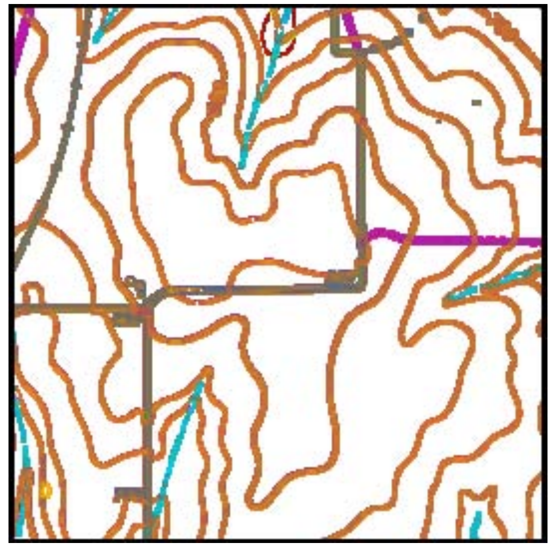

*Figure 6. Eigenvector image created by replacing each pixel in the RGB source image (Figure 1)*

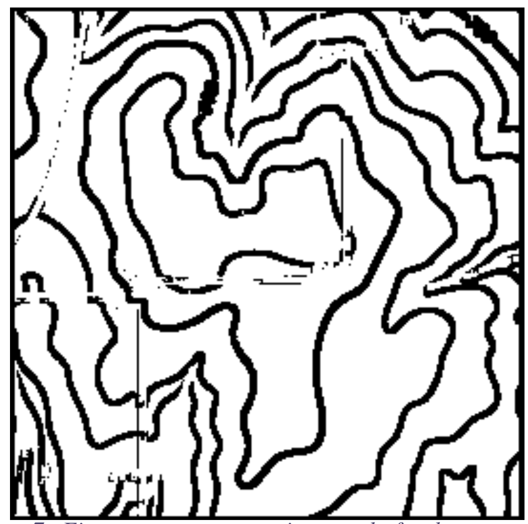

*Figure 7. Eigenvector segmentation result for brown contour lines.*

Next, a smoothing operation is applied to the eigenvector clusters in the spatial domain. A 3 by 3 window is centered about each pixel, and the cluster assignments of these 9 pixels are noted. If six or more pixels have the same pixel assignment, then the center pixel is assigned to the respective cluster. Otherwise, the pixel is assigned to the background cluster. Finally, the white background pixels are identified using RGB clustering of the original image and selecting the largest cluster as shown in Figure 11. All the background pixels are then subtracted from the smoothed eigenvector image. Upon the execution of this stage, the number of clusters is reduced to 11 of which five are significant. These final segmentation results are shown in Figures 12 through 15.

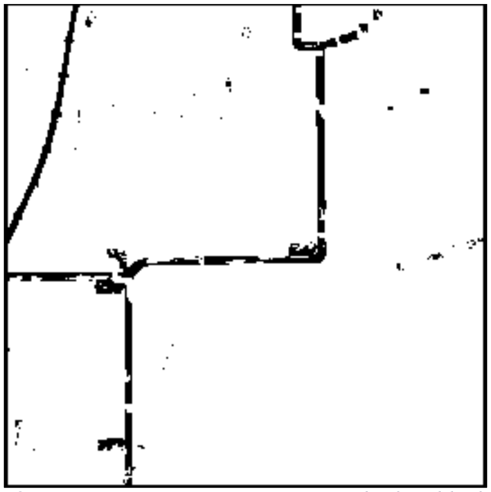

*Figure 8. Eigenvector segmentation result for black pixels representing roads.*

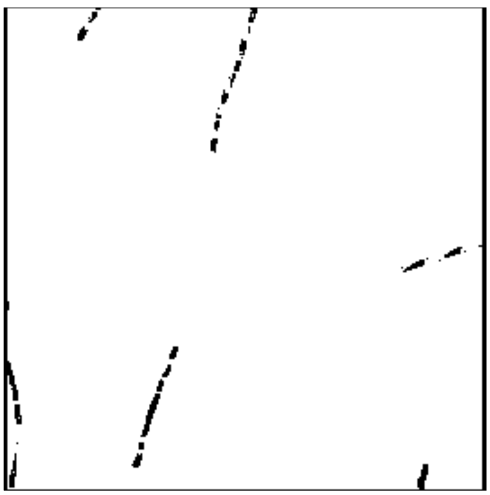

*Figure 9. Eigenvector segmentation result for blue pixels representing rivers.*

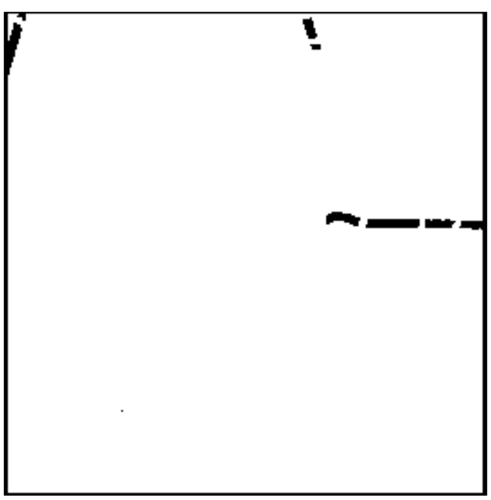

*Figure 10. Eigenvector segmentation result of purple pixels representing map updates.*

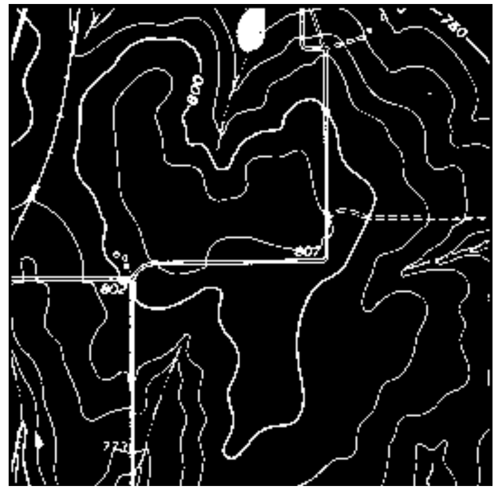


*Figure 11. RGB segmentation for the white background identified by RGB clustering.*

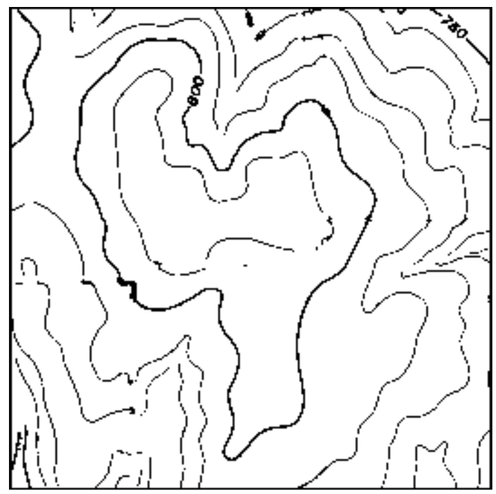


*Figure 12. Final segmentation results for brown contour lines.*

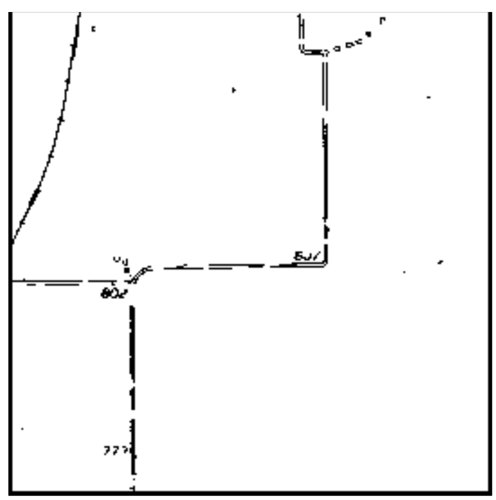

*Figure 13. Final segmentation results for black pixels representing roadways.*

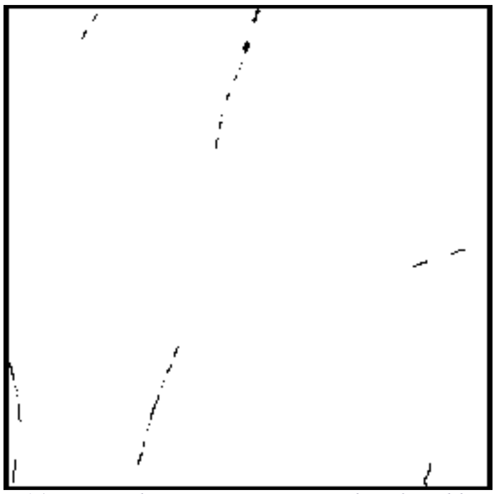

*Figure 14. Final segmentation results for blue pixels representing rivers.*

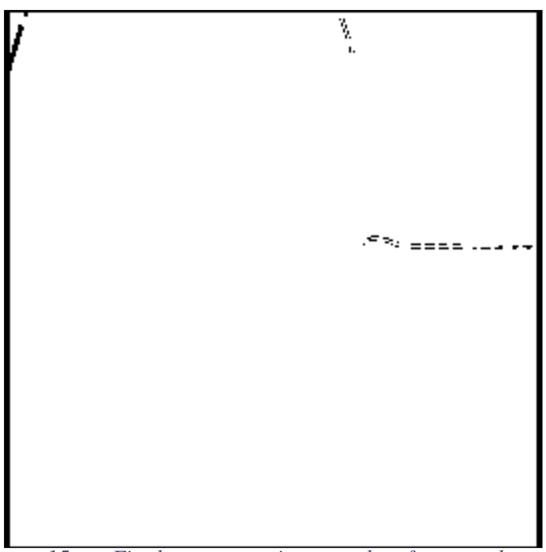

*Figure 15. Final segmentation results for purple pixels representing updates to the map.*

## Results

Inspection of Figures 12 through 15 shows the sample image shown in Figure 1 was automatically segemented into five colors, white background, brown contour lines, black roadways and cultural features, blue rivers and waterways, and purple map updates. This algorithm has been tested on other samples taken from U.S.G.S. paper-based maps with equal success. It has been tested with samples scanned at different resolutions ranging from 100 to 600 dots per inch.

## Conclusions

This algorithm based on eigenvector line-fitting technique in the RGB color domain utilizes both spatial and color information to automatically and accurately segment paper based maps. In all of the experiments, the eigenvector line fitting based color segmentation algorithm performed consistently. It proved to be completely invariant to any RGB misalignments.

## References


1. A. Khotanzad and A. Bouarfa, "Image segmentation by a parallel, non-parametric histogram based clustering algorithm," *Pattern Recognition Journal*, **23**(9), 961-973, (1990).
2. M. Hedley and H. Yan, "Segmentation of color images using spatial and color space information," *Journal of Electronic Imaging*, **1**(4), 374-380 (1992).
3. G. Marcu and S. Abe, "Three-dimensional histogram visualization in different color spaces and applications," *Journal of Electronic Imaging*, **4**(4), 330-346 (1995).
4. R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, Wiley-Interscience, 332-335 (1973). H. Wang and H. Yan, "Text extraction from color map images," *Journal of Electronic Imaging*, **4**(4), 390-396 (1995).
5. T. Uchiyama and M. Arbib, "Color image segmentation using competitive learning," *IEEE Transactions of Pattern Analysis and Machine Intelligence*, **16**(12), 1197-1206 (1994).
6. F. Jenkins, *Fundamentals of Optics*, McGraw-Hill, 176-182 (1976).
7. R. Kasturi, S. Bow, W. El-Masri, J. Shah, J. Gattiker, and U. Mokate, "A system for interpretation of line drawings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12** (10), 978-992 (1990).