

Multilevel Image Thresholding Using the Complement Feature

Salah Ameer

Ontario Colleges, Canada

Article history

Received: 11-06-2020

Revised: 15-08-2020

Accepted: 27-08-2020

Email: salahameer@alumni.uwaterloo.ca

Abstract: A novel Eigen formulation is proposed for image segmentation. Each pixel is represented by a unit vector having the x-component as the normalized gray value of the pixel. The axes of inertia are simply the Eigen vectors of the auto-correlation matrix. The largest Eigen vector is used as the point of split. Each sub-image is further split in the same way. The ratio of the smallest Eigen value to the sum of Eigen values represents the percentage of the minority and was used to control further splitting. The process continues until no sub-image is larger in size than this ratio. The results are very encouraging on a wide range of images.

Keywords: Image Thresholding, Image Segmentation, Eigen Value

Introduction

Image segmentation plays a vital role in almost all image processing and computer vision tasks. The ultimate goal is to delineate the image in such a way to obtain useful descriptions of the objects comprising the scene. To achieve this goal, many algorithms have been (and still being) developed (Goh *et al.*, 2018). Details regarding categorization of these algorithms and the feature space used can be found in many traditional survey papers such as (Sezgin and Sankur, 2004). In fact, the field is so vast and diverse that there are survey papers on a single subcategory e.g., (Oliva *et al.*, 2019; Lucchese and Mitra, 2001; Dey *et al.*, 2010; Ilea and Whelan, 2011; Peng *et al.*, 2013; Unnikrishnan and Hebert, 2005).

In this study, the image segmentation problem is considered as a multilevel thresholding task. A simple but effective Eigen structure is proposed as a solution.

Method

This research is an extension to the work of (Ameer, 2020) on image thresholding using the complement feature. Without loss of generality, the original image is normalized to the interval [0,1] and concatenated to produce a column vector of size $N \times 1$, N is the number of pixels in the image. For the current work, each pixel is represented by one feature, namely *intensity* g_i . Pixel representation is then extended to a 2D *unit* vector having the intensity value as the x-component. In other words, each element is now represented by the vector:

$$G_i = \begin{bmatrix} g_i \\ \sqrt{1-g_i^2} \end{bmatrix} \quad (1)$$

G is now an $N \times 2$ array. An auto correlation matrix (A_G) of size 2×2 are then constructed from G as:

$$A_G = G^T G \quad (2)$$

Solving the Eigen formula:

$$A_G V = \lambda V \quad (3)$$

The Eigen vectors of A_G represent the axes of inertia for the data set. The largest vector V_{\max} (the one corresponding to the maximum Eigen value λ_{\max}) points toward the direction of maximum inertia. Since the y-component is not an independent component, the x-component represents a point of concentration of the original data, close to the mean. The smallest Eigen vector V_{\min} (the one corresponding to the minimum Eigen value λ_{\min}), on the other hand, is normal to V_{\max} and hence, is not guaranteed to have its x-component within the original data range.

The hypothesis adopted in this study can be stated in an abstract form as: The largest eigen value λ_{\max} should be associated with the major process in the data, while the smallest eigen value λ_{\min} should be associated with the minor process. However, no claim is made as whether the major process is the foreground or the background. In fact, it's a mixture of both as will be demonstrated shortly.

This motivates the author to use the x-component of V_{\max} as a threshold. Indeed, the image data is far from being uniform, hence, the previous splitting (thresholding) procedure is successively applied to the newly generated sub-images. In fact, using the mean

instead of the x-component can be used with minor degradation in performance (mean square error).

Another feature of AG worth noting is that its Eigen values, more specifically their ratio, gives a good indication of the percentage of minorities and majorities. Minorities can relate to noise or small objects in reference to the dominant or major object in the image. For this purpose, let's define the Eigen ratio:

$$r_\lambda = \frac{\lambda_{\min}}{\lambda_{\min} + \lambda_{\max}} \quad (4)$$

The splitting procedure, described earlier, will continue as long as the relative size (compared to the original image) is larger than r_λ , the Eigen ratio. For each resulting sub-image, Equation (3) is applied to extract the x-component of V_{\max} .

After completing the splitting procedure, a merging stage can be added to combine adjacent sub-images that are small enough. The merging process is left for a future work. Each resulting region can then be replaced by the x-component of V_{\max} or just the mean.

The full procedure is presented in sequence in Table 1. Hence, four schemes will be tested:

- Scheme PosVec: The x-component of V_{\max} is used as the threshold for each sub-region. Image is normalized to [0,1]
- Scheme NegVec: Similar to scheme PosVec with image normalized to [-1,1]
- Scheme PosMean: Similar to scheme PosVec with mean used as the threshold

- Scheme NegMean: Similar to scheme PosMean with image normalized to [-1,1]

The performance is assessed through the traditional Root Mean Square Error (RMSE) given by:

$$RMSE = \sqrt{\frac{1}{||x||} \sum_{m,n} (x_{mn} - y_{mn})^2} \quad (5)$$

where, x and y stand for original and segmented images and $||x||$ is the cardinality of the set. Adjustment should be placed when the range of images are different. In addition, it is unfair to compare performance between images having different number of segments.

Another evaluation scheme is the SSIM given by, (Wang *et al.*, 2004):

$$SSIM = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \frac{2\sigma_{xy} + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \quad (6)$$

where, μ is the mean, σ^2 is the variance, $c_1 = 0.0001$, $c_2 = 0.0009$ and σ_{xy} is the covariance between x and y .

Experimental Results

The proposed algorithms are implemented on various images shown in Fig. 1.

Results for scheme PosVec, NegVec, PosMean and NegMean are demonstrated in Fig. 2 to 5 respectively. Table 2 to 3 lists the values of Equation (5) and (6) for the results obtained in these schemes.

Table 1: Procedure of the proposed algorithm

1	Read image	
2	Normalize and concatenate.	
3	Append to create a unit vector.	Equation (1)
4	Find autocorrelation matrix.	Equation (2)
5	Find Eigen values and Eigen vectors of autocorrelation matrix.	Equation (3)
6	Find r_λ .	Equation (4)
7	Split image data at the x-component of V_{\max} .	
8	Repeat steps 4–7 for each sub-image occupying more than r_λ of the original image.	
9	Each resulting sub-image will be represented by a uniform intensity.	
10	Small sub-images can be merged.	

Table 2: RMSE for the results of schemes PosVec, NegVec, PosMean and NegMean, Fig. 2 to 5

Image/method	1	2	3	4	5	6
PosVec	0.020	0.019	0.022	0.013	0.030	0.016
NegVec	0.105	0.045	0.067	0.036	0.094	0.032
PosMean	0.018	0.015	0.027	0.014	0.028	0.015
NegMean	0.064	0.039	0.064	0.034	0.051	0.028

Table 3: SSIM for the results of schemes PosVec, NegVec, PosMean and NegMean, Fig. 2 to 5

Image/method	1	2	3	4	5	6
PosVec	0.997	0.995	0.995	0.997	0.995	0.996
NegVec	0.917	0.974	0.958	0.978	0.952	0.985
PosMean	0.997	0.997	0.993	0.996	0.995	0.996
NegMean	0.971	0.981	0.962	0.980	0.986	0.987

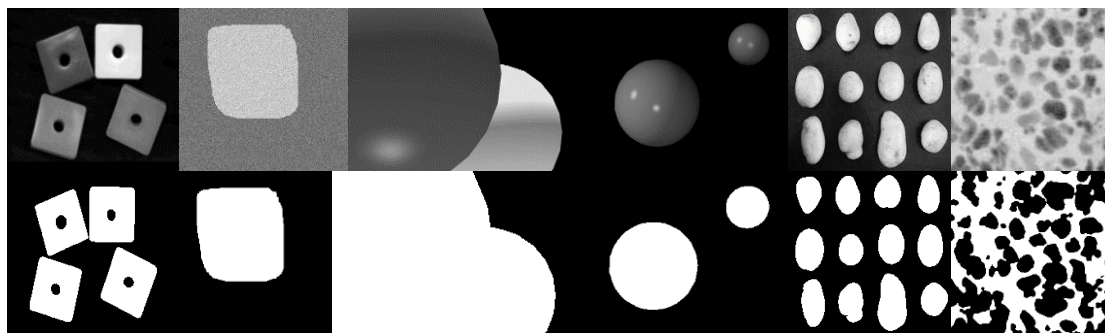


Fig. 1: Test images used and their ground truth

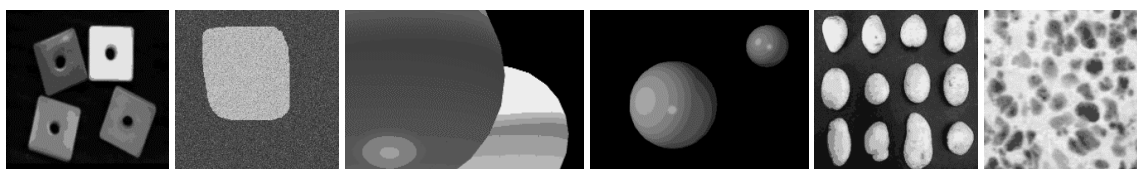


Fig. 2: Segmentation using scheme PosVec (left to right) with number of segments = 14, 24, 16, 10, 10 and 20 respectively



Fig. 3: Segmentation using scheme NegVec (left to right) with number of segments = 6, 9, 5, 3, 4 and 8 respectively

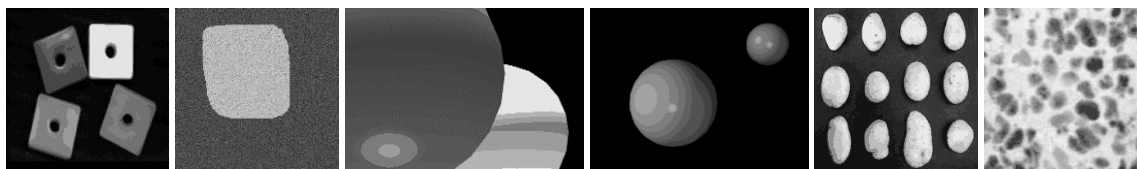


Fig. 4: Segmentation using scheme PosMean (left to right) with number of segments = 14, 24, 16, 10, 10 and 21 respectively

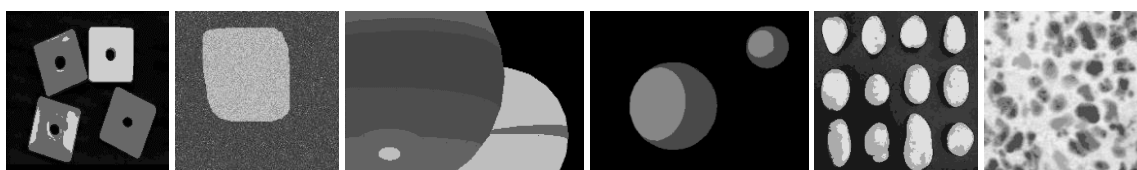


Fig. 5: Segmentation using scheme NegMean (left to right) with number of segments = 6, 10, 5, 4, 5 and 10 respectively

It can be seen from Fig. 2 to 5 and Table 2 and 3 that scheme PosVec and PosMean are similar except for the computational cost favouring PosVec. Since finding the Eigen values is required by both schemes PosVec and PosMean, the extra step may be in favour of PosVec as the matrix is already there. On the other hand, schemes NegVec and NegMean produce less number of regions.

It should be emphasised that the values of RMSE and SSIM have to be used in comparing images with the same number of regions. The values of RMSE and

SSIM clearly demonstrate the effectiveness of the proposed schemes.

To further illustrate the effectiveness of the proposed schemes, some images from Berkley Segmentation Database (BSD) (Martin *et al.*, 2001) are segmented in Fig. 6 and 7. As the values of RMSE and SSIM are not good discriminators for high quality segmentation, they will not be reported for these images. It is easily noticed from these images that using the normalization $[-1,1]$ produces less number of regions compared to $[0,1]$.

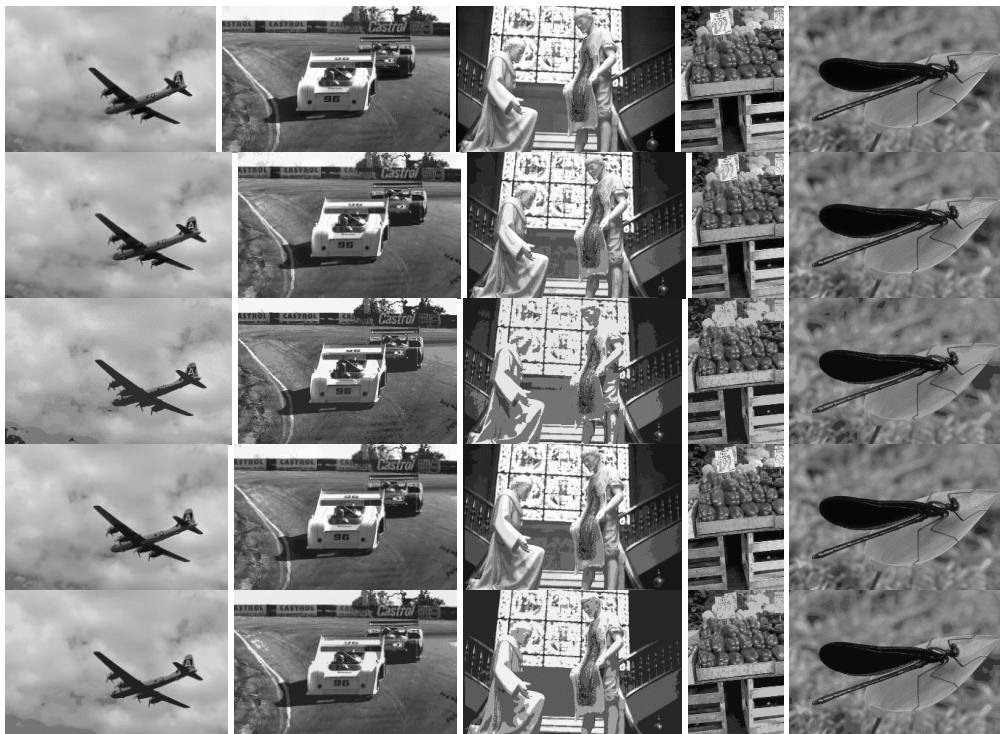


Fig. 6: (Top to bottom) Original images from BSD followed by segmentations using schemes PosVec, NegVec, PosMean and NegMean with their respective number of segments: 3096(53, 21, 51, 21), 21077(16, 7, 17, 7), 24077(9, 4, 9, 4), 25098(17, 6, 17, 7) and 35070(50, 12, 50, 13)

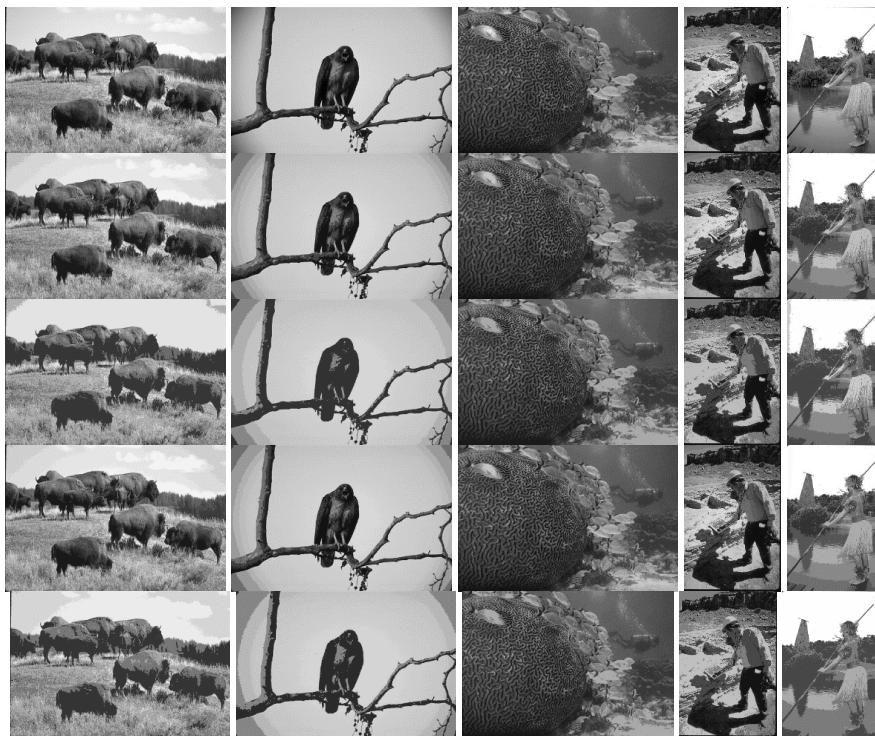


Fig. 7: (Top to bottom) Original images from BSD followed by segmentations using schemes PosVec, PosMean and NegMean with their respective number of segments: 38092(15, 5, 15, 5), 42049(20, 9, 20, 8), 45077(54, 14, 54, 14), 89072(18, 5, 18, 5) and 101087(7, 5, 8, 4)

However, the latter has better performance in preserving majority of the details without diminishing small regions that are of visual importance to the viewer. Despite the fact that mean can be used, the x -component of V_{\max} can be of better computational benefit as the Eigen values are already computed.

Conclusion and Future Work

A simple Eigen structure is proposed in this study to perform multi-level image segmentation using one feature, intensity. The algorithm is fully automatic and no adjustment to any sort of parameters is needed.

The proposed schemes are very effective as demonstrated by the values of RMSE and SSIM. However, better evaluation schemes are needed to better compare the performances.

Work is currently in progress to extend the algorithm to segment colored images. Other feature spaces, specifically ones incorporating local information, need to be investigated to improve segmentation quality.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

- Ameer, S. (2020). Image Thresholding using the Complement Feature. *American Journal of Engineering and Applied Sciences*, 13, 311-317.
- Dey, V., Zhang, Y., & Zhong, M. (2010). A review on image segmentation techniques with remote sensing perspective (Vol. 38, pp. 31-42). Vienna, Austria: na.
- Goh, T. Y., Basah, S. N., Yazid, H., Safar, M. J. A., & Saad, F. S. A. (2018). Performance analysis of image thresholding: Otsu technique. *Measurement*, 114, 298-307.
- Ilea, D. E., & Whelan, P. F. (2011). Image segmentation based on the integration of colour–texture descriptors-A review. *Pattern Recognition*, 44(10-11), 2479-2501.
- Lucchese, L., & Mitra, S. K. (2001). Colour image segmentation: a state-of-the-art survey. *Proceedings-Indian National Science Academy Part A*, 67(2), 207-222.
- Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001, July). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001* (Vol. 2, pp. 416-423). IEEE.
- Oliva, D., Abd Elaziz, M., & Hinojosa, S. (2019). Multilevel thresholding for image segmentation based on metaheuristic algorithms. In *Metaheuristic Algorithms for Image Segmentation: Theory and Applications* (pp. 59-69). Springer, Cham.
- Peng, B., Zhang, L., & Zhang, D. (2013). A survey of graph theoretical approaches to image segmentation. *Pattern recognition*, 46(3), 1020-1038.
- Sezgin, M., & Sankur, B. (2004). Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic imaging*, 13(1), 146-166.
- Unnikrishnan, R., & Hebert, M. (2005, January). Measures of similarity. In *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05)-Volume 1* (Vol. 1, pp. 394-394). IEEE.
- Wang, A., C Bovik, H., Sheikh, R., & Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4), 600-612.