Journal of Engineering and Technology for Industrial Applications

ITEGAM-JETIA

ISSN ONLINE: 2447-0228

Manaus, v.7 n.31, p. 11-17. Sept/Oct, 2021
DOI: https://doi.org/10.5935/jetia.v7i31.777

RESEARCH ARTICLE OPEN ACCESS

A CLOSED-FORM SOLUTION TO MULTI-LEVEL IMAGE THRESHOLDING

Salah Ameer*[1]

[1] Electronics Engineering, Conestoga College, Kitchener, ON, Canada.

[1] http://orcid.org/0000-0001-7930-6691

Email: *salahameer@alumni.uwaterloo.ca

ARTICLE INFO

Article History
Received: October 20th, 2021
Accepted: October 26th, 2021
Published: October 29th, 2021

Keywords:
Image Thresholding,
OTSU,
Multi-level,
Non-exhaustive Search.

ABSTRACT

This paper proposes an analytical formulation relying on the least square error. Similar results were also found for the cross correlation, within-class, and between-class variance. At first, a continuous distribution is hypothesized (for derivation purposes only) to produced a modified form of the well-known OTSU method. This hypothesis is “identical” to Otsu in terms of output performance and the need for an exhaustive search. However, apart from being derived from the continuous form, the proposed scheme requires less computational power. It turns out that the optimum threshold equals the average of the adjacent regions’ means. For some images, the scheme can result in multi-level thresholdeds. A direct form was then suggested to obtain a non-exhaustive solution. The idea is simply to approximate the non-continuous error function (used by the least square formulation and OTSU) with a forth order polynomial defined in the normalized gray intensity range [0,1]. The optimum threshold can then be found as a function of the roots of a second order polynomial whose coefficients are the solution of a 2x2 linear system. The performance of the proposed non-exhaustive solution is slightly inferior to OTSU in general, however; some images produced improved performance. Nevertheless, The proposed scheme can be easily generalized to the multi-level case without the need for an exhaustive search. For n+1 levels (i.e. n thresholds), the output is obtained by solving an nxn linear system followed by finding the roots of a n-order polynomial. The computational cost is clearly superior to the exhaustive search. In addition, as validated with some images, the performance is encouraging. Extension to the general clustering case is highly envolved with the exception of the two-level case (for any dimension) that has been successfully derived in this work.

I. INTRODUCTION

Image thresholding is vital in many applications as it allows to separate objects in an image, typically based on their intensity. Various schemes have been proposed in the literature, a good review can be found in [1].

The histogram plays a crucial role in many of these schemes. Many schemes (e.g. [2], [3]) use the histogram as an approximation to a probability density function. An objective function is then formalized dependent on some features or attributes of the histogram, such as variance and entropy. The threshold is then selected as a solution to optimize this objective function.

Due to the fact that a histogram does not carry spatial information (2 different images can have the same histogram), higher dimensional histograms have been proposed [4-6].

The aforementioned schemes can be generalized to multi-level thresholding [7]. However, the computational price is too high. In addition, having many thresholds, the ensemble size for each region is reduced. This often results in inferior quality since statistics (or probability distribution, i.e., histogram) rely heavily on a large ensemble size [8].

There are many measures [1, 8-9] to evaluate the performance of a thresholding scheme. However, application dependant, a subjective decision may be preferred.

II. MATERIALS AND METHODS

II.1 ANALYTICAL FORMULATION

Consider the minimization of the within-class variance [2]

$$V_w = w_1\sigma_1^2 + w_2\sigma_2^2 \tag{1}$$

Where index 1(2) represent a parameter belonging to region 1(2) respectively, w is weight, m is mean, h is histogram, and σ^2 is variance.

Substuting their definitions according to [2] (with integral replacing summation), we have

$$V_w = w_1\int_0^T (t-m_1)^2\frac{h(t)}{w_1}dt + w_2\int_T^1 (t-m_2)^2\frac{h(t)}{w_2}dt$$

$$V_w = \int_0^T t^2h(t)dt - w_1m_1^2 + \int_T^1 t^2h(t)dt - w_2m_2^2$$

The combined integral in the above equation is independent of T, hence, the problem is to maximize

$$F_w = \frac{\left\{\int_0^T th(t)dt\right\}^2}{\int_0^T h(t)dt} + \frac{\left\{\int_T^1 th(t)dt\right\}^2}{\int_T^1 h(t)dt} \tag{2}$$

Setting the derivative w.r.t T to zero and avoiding zeros in the histogram, we obtain

$$\frac{2T\left\{\int_0^T h(t)dt\right\}\left\{\int_0^T th(t)dt\right\} - \left\{\int_0^T th(t)dt\right\}^2}{\left\{\int_0^T h(t)\right\}^2} = \frac{2T\left\{\int_T^1 h(t)\right\}\left\{\int_T^1 th(t)dt\right\} - \left\{\int_T^1 th(t)dt\right\}^2}{\left\{\int_T^1 h(t)\right\}^2}$$

The above result can be simplified using the same definitions used in (1) to

$$2Tm_1 - m_1^2 - 2Tm_2 + m_2^2 = 0$$

$$(m_2 - m_1)\{m_2 + m_1 - 2T\} = 0$$

$$m_1 + m_2 = 2T \tag{3}$$

As region means cannot be equal, the simplification in (3) was obtained. The result is still an exhaustive search, the procedure will be called Optimum Exhaustive (OE) henceforth. It can be clearly noticed that the computational burden is less than that required by the OTSU method.

An interesting observation of (1) is that the formulation can be modified to produce the least square error (LSE) between the original and the thresholded image. Hence, (3) and OTSU method can both be categorized as least square solutions. In other words, the objective function to minimize is actually

$$V_w = \int_0^T (t-m_1)^2h(t)dt + \int_T^1 (t-m_2)^2h(t)dt \tag{4}$$

The same result (with more elaborate mathematics) can be obtained using the between-class variance [2],

$$V_b = w_1w_2(m_2 - m_1)^2 \tag{5}$$

In fact, same result (using a similar procedure) can be obtained by minimizing the cross correlation between the histograms of the original image and that of the thresholded image, i.e. using the formulation

$$F_{Cross} = \frac{X_1 + X_2}{V_O V_T} \tag{6a}$$

$$V_O = \sqrt{\int_0^1 \{x - m\}^2 h(x)dx} \tag{6b}$$

$$V_T = \sqrt{\int_0^T \{m_1 - m\}^2h(x)dx + \int_T^1 \{m_2 - m\}^2h(x)dx} \tag{6c}$$

$$X_1 = \int_0^T \{x - m\}\{m_1 - m\}h(x)dx \tag{6d}$$

$$X_2 = \int_T^1 \{x - m\}\{m_2 - m\}h(x)dx \tag{6e}$$

Interestingly, the complement feature formulation recently proposed [10] produces the same outcome as in (3). The complement feature is implemented in this work as the dot product between the original image and the thresholded image, i.e.

$$F_{Comp} = \int_0^T \{xm_1 + (1-x)(1-m_1)\}h(x)dx + \int_T^1 \{xm_2 + (1-x)(1-m_2)\}h(x)dx \tag{7}$$

Other features (e.g., entropy) can be treated in the same fashion as (1), however, the resultant description for the threshold is not as tractable as in (3).

The proposed scheme can be extended to multilevel thresholding resulting in an extended formulation to that in (3). In fact, (2) can be extended to multi-level as

$$F_n = \sum_{i=1}^{n} \frac{\left\{\int_{T_{i-1}}^{T_i} th(t)dt\right\}^2}{\int_{T_{i-1}}^{T_i} h(t)dt} \tag{8}$$

Following a similar derivation as above but more than one threshold, we obtain (please note that the number of means is more than the number of thresholds by one)

$$2T_jm_j - m_j^2 - 2T_jm_{j+1} + m_{j+1}^2 = 0$$

$$2T_j = \frac{m_{j+1}^2 - m_j^2}{m_{j+1} - m_j} = m_{j+1} + m_j \tag{9}$$

Unfortunately, the search is still an exhaustive one.

II.2 NON-EXHAUSTIVE SOLUTIONS

Despite the simple formulations obtained for two- and multi-level thresholding, as shown in (3) and (9) respectively, the solution can only be obtained through an exhaustive search.

Resulting in a computational cost almost exponentially dependent on the number of levels.

A closer look at (4) reveals the discontinuity of the error function implemented. Hence, continuous approximations to (4) will be investigated in this subsection. Polynomials are used to find a direct solution and further extend it to the multi-level case.

The essential idea is to reformulate (4) as

$$F = \int_0^1 S(t, m_1, m_2) h(t) dt \qquad (10)$$

Where

$$S(t, m_1, m_2) \approx \begin{Bmatrix} (t-m_1)^2 & t < T \\ (t-m_2)^2 & t > T \end{Bmatrix} \qquad (11)$$

One of the possibilities for *S* is shown in Figure 1. An important requirement to overcome the need for an exhaustive search is to force *S* to be a continuous function.

There are many ways to have a continuous representation for *S*. One suggestion would be to have a polynomial optimizing the objective function

$$F = \int_0^T \left\{ \sum_{i=0}^{n} b_i (t-T)^i - (t-m_1)^2 \right\}^2 dt + \int_T^1 \left\{ \sum_{i=0}^{n} b_i (t-T)^i - (t-m_2)^2 \right\}^2 dt \qquad (12)$$

Where T is given by (3). Some simplifications can be used to ease the elaborate mathematics, however, a 7th order polynomial needs to be solved for n=4 (order of *S*). Unfortunately, the description in (12) cannot be easily modified to the multi-level case.

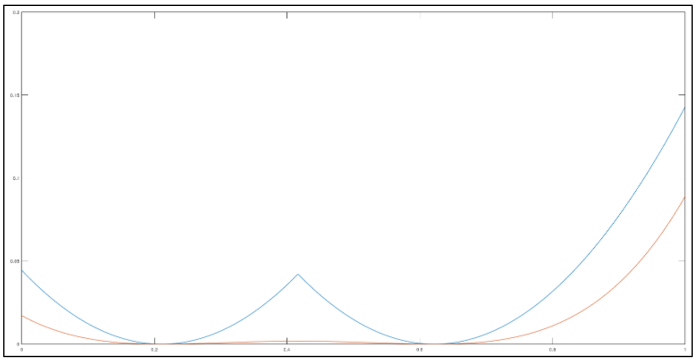

Figure 1: (blue) A possible plot for the function *S* defined in (11) and (red) a 4th order approximation.
Source: Author, (2021).

An easier and powerful (as will be shown later) solution would be to concentrate on special features of *S* rather than the global approximation. The most attracting features of *S* are its zeros that are also its minima. The simplest form to describe this behavior is the quadratic product (QP) shown as

$$S_2(t, m_1, m_2) = (t-m_1)^2 (t-m_2)^2 \qquad (13)$$

The description in (13) has deviation all over the range ($t \in [0,1]$), (see Figure 1). This sacrifice enables for a more regourous solution of (10) as

$$F_{QP} = \int_0^1 (t-m_1)^2 (t-m_2)^2 h(t) dt \qquad (14)$$

Setting the derivatives in (14) w.r.t m_1 and m_2 to zero results in

$$\frac{\partial F_{QP}}{\partial m_1} = \int_0^1 (t-m_1)(t-m_2)^2 h(t) dt = 0 \qquad (15a)$$

$$\frac{\partial F_{QP}}{\partial m_2} = \int_0^1 (t-m_1)^2 (t-m_2) h(t) dt = 0 \qquad (15b)$$

Subtracting (15a) from (15b) results in

$$\int_0^1 (t-m_1)(t-m_2)(m_2-m_1) h(t) dt = 0 \qquad (16)$$

Substituting (16) back into (15a), we have

$$\int_0^1 (t-m_1)(t-m_2) t h(t) dt = 0 \qquad (17a)$$

$$\int_0^1 (t-m_1)(t-m_2) h(t) dt = 0 \qquad (17b)$$

Define

$$y_i = \int_0^1 t^i h(t) dt \qquad (18)$$

Using (18), we can rewrite (17) using polynomial notation as

$$y_0 P_0 + y_1 P_1 = -y_2 \qquad (19a)$$

$$y_1 P_0 + y_2 P_1 = -y_3 \qquad (19b)$$

Where P_i are the coefficients of a second order polynomial obtained from (17). Explicitly,

$$P_0 = m_1 m_2 \qquad (20a)$$

$$P_1 = -(m_1 + m_2) \qquad (20b)$$

The 2x2 linear system of (19) can now be solved for P_i. The solutions (P_1 and P_2) can then be used to find the two centers m_1 and m_2 as the roots of the second order polynomial given by

$$P_0 + P_1 x + x^2 = 0 \qquad (21)$$

It should be clarified that P_i are used to find m_i without the explicit use of (20). This trick enables the procedure to be break into two simple subsystems (a linear system followed by a root-finding scheme) rather than solving a nonlinear set of equations.

The threshold is then found as the average of the centres m_i. Note the similarity with (3).

A more intuitive look at Figure 1 suggests that the threshold is the maxima of the polynomial (minima are the centres m_i). Hence, the derivative of (13) is

$$\frac{\partial}{\partial t}\{(t-m_1)^2 (t-m_2)^2\} = 0$$

$$(t-m_1)^2 2(t-m_2)+2(t-m_1)(t-m_2)^2=0$$

$$(t-m_2)(t-m_1)\{t-m_1+t-m_2\}=0 \quad (22)$$

The first two terms are simply the minima and are of no use as a threshold. Hence, the useful term is the last one. To reduce computation, (22) can be directly obtained from the derivative of (21) as

$$2x+P_1=0 \quad (23)$$

Table 1: Procedure flow of QP2.

#	Description
1	Find Image Histogram
2	Evaluate (18)
3	Solve 2x2 system (19)
4	Solve (23) to find threshold

Source: Author, (2021).

Interestingly, the solution obtained by (23) is the same as that given by (3). This clearly supports the optimum behaviour of the choice given in (13). Hence, the procedure relying on (14) will be called Quadratic Product for level 2 (QP2), see Table 1. The number 2 in QP2 is to indicate that the output is two-level (one threshold).

II.3 VARIATIONS TO (12)

A closer look at Figure 1 reveals the feature points of the function at 0, m_1, T, m_2, and 1. Satisfying the conditions (the function and/or the derivative) imposed by one or more of these feature points results in a higher order polynimial for S in (10). Unfortunately, the polynimial coefficients are also nonliear in terms of the unknowns m_1 and m_2. Despite the fact that finding the roots of a high order may be preferred over exhaustive search, the description does not lend itself to be generalized to multi-level thresholding.

One of the interesting formulations was shown in (14). All other trials were too complex except ones that are just a multiplication of (13) by a constant in the form

$$S_c(t,m_1,m_2)=C(T,R)(t-m_1)^2(t-m_2)^2 \quad (24a)$$

$$2T=m_2+m_1 \quad (24b)$$

$$2R=m_2-m_1 \quad (24c)$$

Many forms similar to (24a) can be designed to produce various schemes with different level of complexty. One such scheme is

$$F_{Ratio}=\int_0^1\left(\frac{t}{m_1}-1\right)^2\left(\frac{t}{m_2}-1\right)^2 h(t)dt \quad (25)$$

Leading to the solution (in a similar fashion to DS2)

$$\begin{bmatrix} y_1 & y_2 \\ y_2 & y_3 \end{bmatrix}\begin{bmatrix} P_0 \\ P_1 \end{bmatrix}=-\begin{bmatrix} y_3 \\ y_4 \end{bmatrix} \quad (26)$$

It was noted that (26) can give slightly better results as compared to QP2. However, due to the denominators, the previous statement is valid either to the image or its negative (1 – image) but not both. Leading to a significant drawback for the multi-level case.

Many formulations of (24) have been investigated and found to produce solutions requiring the root finding of polynomials (in T) of order >6.

The function given in (13) can be further modified using just a multiplication constant. The simplest approaches were to exactly satisfy (11) at either t=0, T, 1, 1+T, or 1–T. Unfortunately, these suggestions were either mathematically involved or produce inferior performance (inefficient for some images). Another drawback is the inability of these schemes to generalize to the multi-level case.

The function S, given in (10), can be described using a polynomial of high order (>4) to better satisfy (11). Unfortunately, the resultant mathematics was found to be more involved for many suggestions including the ones described in the previous paragraph.

II.4 MULTI-LEVEL THRESHOLS

Another advantage of the suggestion in (13) is that it can be easily extended to multi-level thresholding through

$$S_n(t,m_1,m_2)=\prod_{i=1}^{n}(t-m_i)^2 \quad (27)$$

Following a similar procedure to that obtained through (14) to (18), we obtain the following linear system

$$\begin{bmatrix} y_0 & \cdots & y_{n-1} \\ \vdots & \ddots & \vdots \\ y_{n-1} & \cdots & y_{2n-2} \end{bmatrix}\begin{bmatrix} P_0 \\ \vdots \\ P_{n-1} \end{bmatrix}=-\begin{bmatrix} y_n \\ \vdots \\ y_{2n-1} \end{bmatrix} \quad (28)$$

Once P_i are obtained (note the extenstion from (20)), we can find m_i by finding the roots of

$$\sum_{i=0}^{n-1} P_i x^i + x^n = 0 \quad (29)$$

Each threshold is then found by averaging the two adjacent centres in a similar fashion to (9). The same procedure leading to (22) can be extended to simplify the scheme of finding the thresholds as follows,

$$\frac{\partial}{\partial x}\prod_{i=1}^{n}(x-m_i)^2=0$$

$$\sum_{j=1}^{n}\prod_{i=1}^{n}\frac{2(x-m_i)^2}{(x-m_j)}=0 \quad (30)$$

Excluding the minima as the thresholds are the maxima (see Figure 1), we have

$$\sum_{j=1}^{n}\prod_{\substack{i=1\\ i\neq j}}^{n}(x-m_i)=0 \quad (31)$$

Equivalently, (31) can be written as

$$\frac{\partial}{\partial x}\prod_{i=1}^{n}(x-m_i)=\frac{\partial}{\partial x}\left\{\sum_{i=0}^{n-1}P_i x^i+x^n\right\}=0$$

$$nx^{n-1}+\sum_{i=1}^{n-1} iP_i x^{i-1}=0 \qquad (32)$$

Obviously, the order of the polynomial in (32) is less by one compared to that in (29). In addition, it avoids the avaraging procedure required to find the thresholds from the obtained means m_i. The procedure will be called Quadratic Product for level n (QPn) as descibed in Table 2. Please note that QPn has n–1 thresholds.

Table 2: Procedure flow of QPn.

#	Description
1	Find Image Histogram
2	Evaluate (18)
3	Solve linear system (28)
4	Solve (32) to find thresholds

Source: Author, (2021).

II.5 EXTENSION TO CLUSTERING

Fortunately, a direct solution to the two-level case (for any number of dimensions) can be formulated as will be given shortly. Raw data will be used (without histogram) to ease the notation. Consider the optimization problem

$$F_{Dim}=\sum_i\left((x_i-m_1)^T(x_i-m_2)\right)^2$$

$$F_{Dim}=\sum_i\sum_{k=1}^{2}(x_{ik}x_{ik}-x_{ik}m_{1k}-x_{ik}m_{2k}+m_{1k}m_{2k})^2 \qquad (33)$$

Let's define

$$a_{jk}=\sum_i x_{ik}^j \qquad (34a)$$

$$b_k=a_{0k}a_{2k}-a_{1k}a_{1k} \qquad (34b)$$

$$c_k=a_{0k}a_{3k}-a_{1k}a_{2k} \qquad (34c)$$

$$d_k=a_{1k}a_{3k}-a_{2k}a_{2k} \qquad (34d)$$

Setting derivatives of F (33) to zero and performing a few simplifications, we obtain

$$b_k m_{2k}=c_k-b_k m_{1k} \qquad (35)$$

$$m_{1k}=\frac{c_k\pm\sqrt{c_k^2-4b_k d_k}}{2b_k} \qquad (36)$$

Please note that each of the equations (34) – (36) are actually a set written compactly together. In other words, for a 3D data, (36) is in fact a set of three equations.

The scheme (QPn) can be extended to levels more than two and higher dimensions (clustering), however, the resultant mathematics is too much envolved. Enforcing independency (sub-optimal) between the dimensions can reduce the complexity on the expense of ambiguity in the final results. The ambiguity manefists itself in the fact that n centers should be picked out of the produced n^k centers (k is the number of dimensions).

III. RESULTS AND DISCUSSIONS

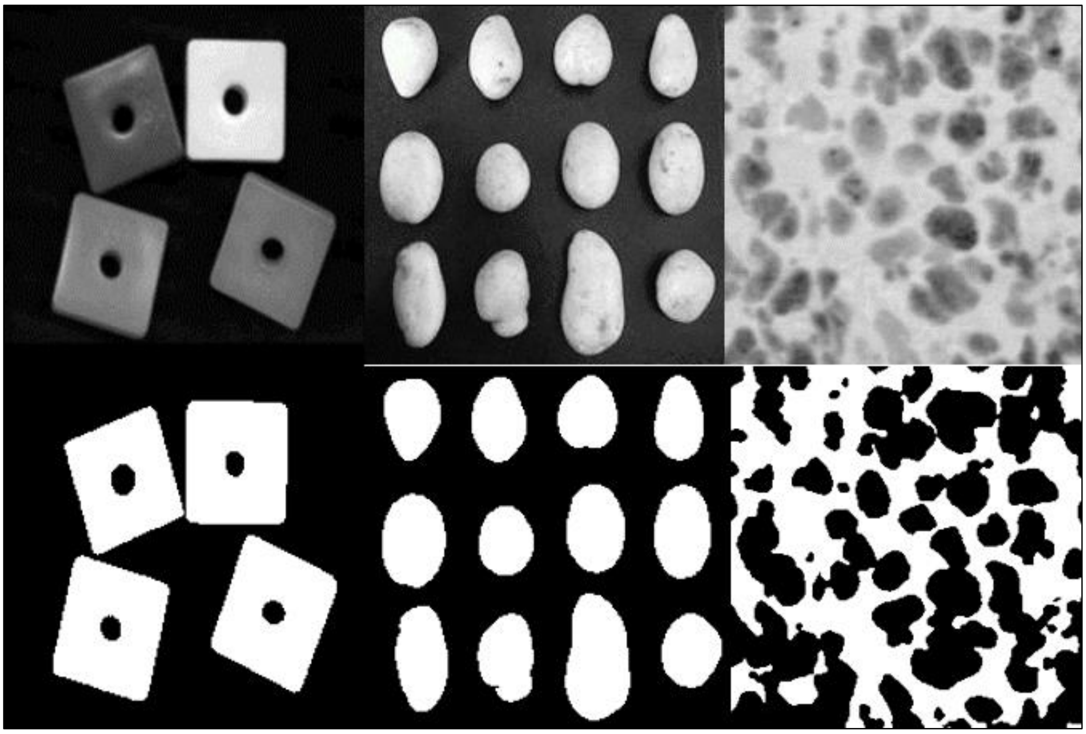

Figure 2: (First row) test images (1 – 3), (second row) their ground truth.
Source: Author, (2021).

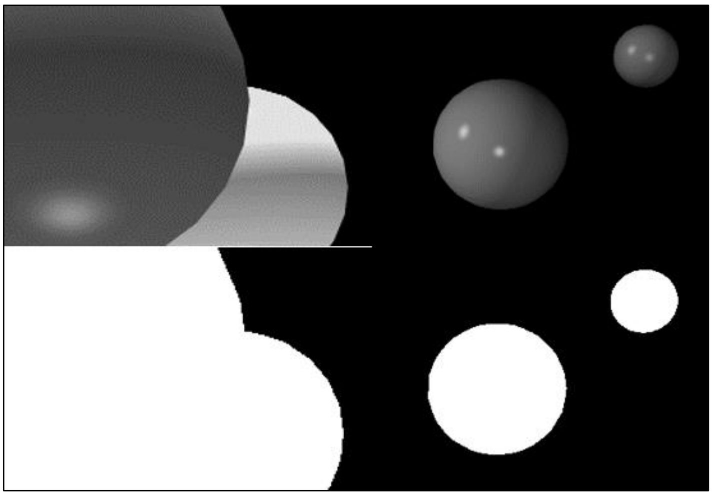

Figure 3: (First row) test images (4 – 5) and (second row) their ground truth.
Source: Author, (2021).

Five test images with their ground truth are shown in Figure 2-3. For comparison purposes, the outputs of OTSU, OE (3), and QP2 (see Table 1) are shown in Figure 4-5.

OE is clearly similar to OTSU. The thresholds are very close, however; for some images, OE produces more than one threshold, an observation that may wort further investigation.

Although the performance of QP2 is inferior to both OTSU and OE (more images are needed to obtain useful statistics for performance comparisons), the non-exhaustive behaviour is appealing. In addition, results for number of levels more than 2 is encouraging. Figure 6-7 shows the performance of QP3, and QP4 using the same set of images.

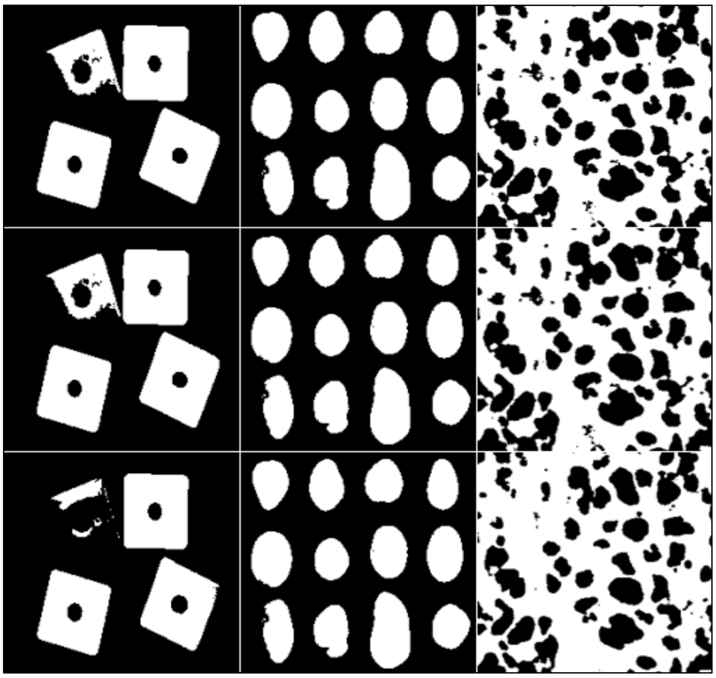

Figure 4: results of OTSU (First row), OE (second row), and QP2 (third row) for the test images (1 – 3).
Source: Author, (2021).

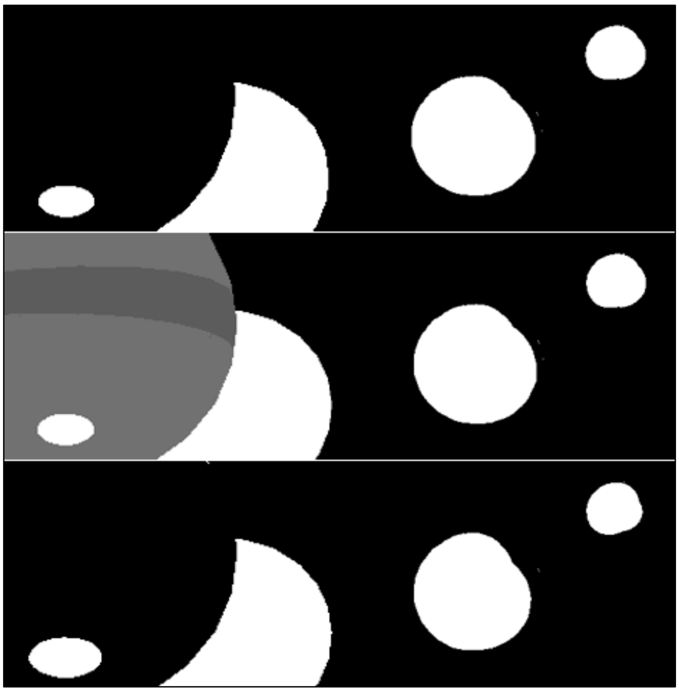

Figure 5: results of OTSU (First row), OE (second row), and QP2 (third row) for the test images (4 – 5).
Source: Author, (2021).

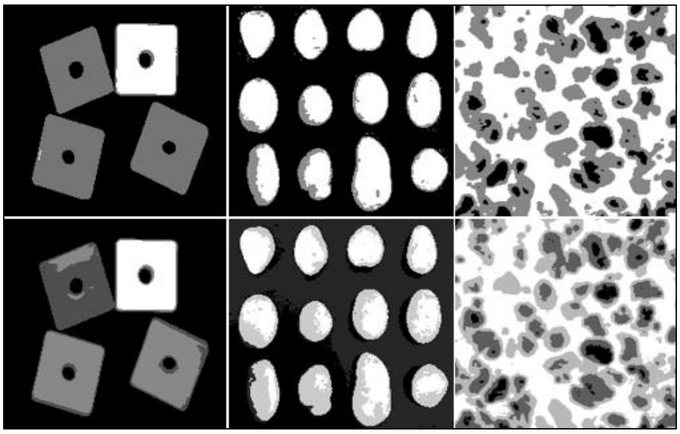

Figure 6: results of QP3 (First row) and QP4 (second row) for the test images (1 – 3).
Source: Author, (2021).

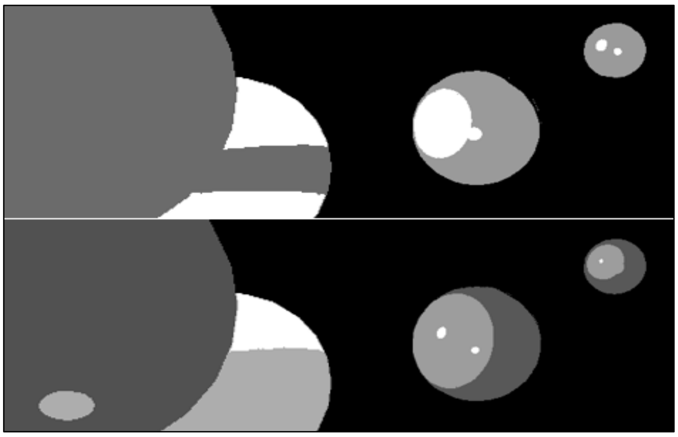

Figure 7: results of QP3 (First row) and QP4 (second row) for the test images (4 – 5).
Source: Author, (2021).

It was noticed that smoothing the histogram (through averaging and/or fitting) did not produce significant differences.

More images where thresholded using QP3 to further explore the performance as shown in Figure 8-9.

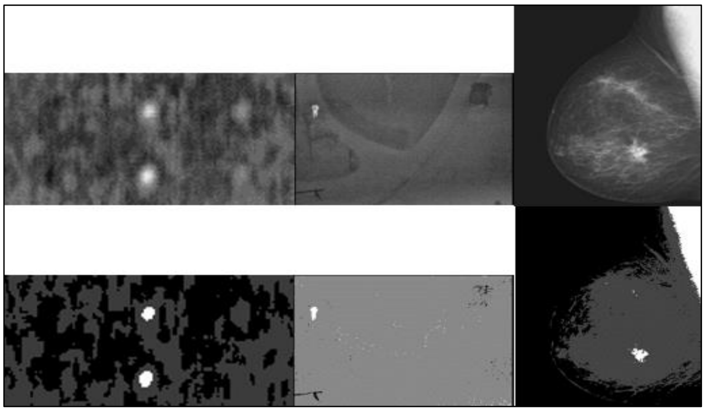

Figure 8: (First row) test images (6 – 8) and (second row) their QP3 output.
Source: Author, (2021).

Figure 9: (First row) test images (9 – 11) and (second row) their QP3 output.
Source: Author, (2021).

IV. CONCLUSIONS

A new optimized solution is suggested for the binary image thresholding problem using least square error. The formulation is equivalent to the commonly used OTSU method and the cross-correlation description. Although performance does not deviate too much from that of OTSU, a computational gain is obtained.

Inspired by the mentioned optimized solution, a non-exhaustive scheme was then developed using a polynomial approximation to the least square error function. A quadratic product was suggested as a general extension.

The quadratic product QP2 has shown encouraging results. In addition to its simplicity, the formulation can be easily extended to the multi-level thresholding case QPn.

Some variations (still non-exhaustive) have been tried without significant improvements in performance but are computationally demanding. Further insight into higher order polynomials may be needed to investigate a possible compromise between performance and complexity.

More weighting schemes, using (24), are currently under investigation to improve performance while keeping the mathematics tractable.

A more elaborate comparison is being conducted to further explore the extent of the proposed schemes in terms of output quality and computational cost.

V. AUTHOR’S CONTRIBUTION

Conceptualization: Salah Ameer.
Methodology: Salah Ameer.
Investigation: Salah Ameer.
Discussion of results: Salah Ameer.
Writing – Original Draft: Salah Ameer.
Writing – Review and Editing: Salah Ameer.
Resources: Salah Ameer.
Supervision: Salah Ameer.
Approval of the final text: Salah Ameer.

VI. REFERENCES

[1] M. Sezgin, and B. Sankur, “Survey over image thresholding techniques and quantitative performance evaluation”, J. Electronic Imag., 13: pp 146-165, 2004. DOI: 10.1117/1.1631315.

[2] N. Otsu, “A threshold selection method from gray-level histograms”, IEEE Trans. Syst. Man Cybernet., 9: pp 62-66, 1979. DOI: 10.1109/TSMC.1979.4310076.

[3] J. Kapur, P. Sahoo and A. Wong, “A new method for gray-level picture thresholding using the entropy of the histogram”, Comput. Vis. Graph. Image Process., 29: 273-285, 1985. DOI: 10.1016/0734-189X(85)90125-2

[4] J. Zhang, and J. Hu, “Image segmentation based on 2D Otsu method with histogram analysis”, Proceedings of the International Conference on Computer Science and Software Engineering, Dec. 2008, IEEE Xplore Press, Hubei, China, pp: 105-108. DOI: 10.1109/CSSE.2008.206.

[5] A. Abutaleb, “Automatic thresholding of gray-level pictures using two-dimensional entropy”, Comput. Vis. Graph. Image Process., 47: 22-32, 1989. DOI: 10.1016/0734-189X(89)90051-0.

[6] X. Zheng, H. Ye and Y. Tang, “Image bi-level thresholding based on gray level-local variance histogram”, Entropy, 19: pp 1-8, 2017. DOI: 10.3390/e19050191.

[7] D. Liu, and J. Yu, “Otsu method and k-means”, Proceedings of the 9th International Conference on Hybrid Intelligent Systems, Aug. 2009, IEEE Xplore Press, Shenyang, China, pp: 344-349. DOI: 10.1109/HIS.2009.74.

[8] S. Ameer, “Histogram Matching Schemes for Image Thresholding”, American Journal of Engineering and Applied Sciences, pp 413 – 419, 2019. https://thescipub.com/pdf/ajeassp.2019.413.419.pdf

[9] S. Ameer, “An Eigen Structure Involving the Histogram for Image Thresholding”, IET-Image Processing, 2020, pp 1–4. **DOI:** 10.1049/iet-ipr.2019.1428.

[10] S. Ameer, “Image Thresholding Using the Complement Feature”, American Journal of Engineering and Applied Sciences, 2020, pp 311 – 317. https://thescipub.com/pdf/ajeassp.2020.311.317.pdf.