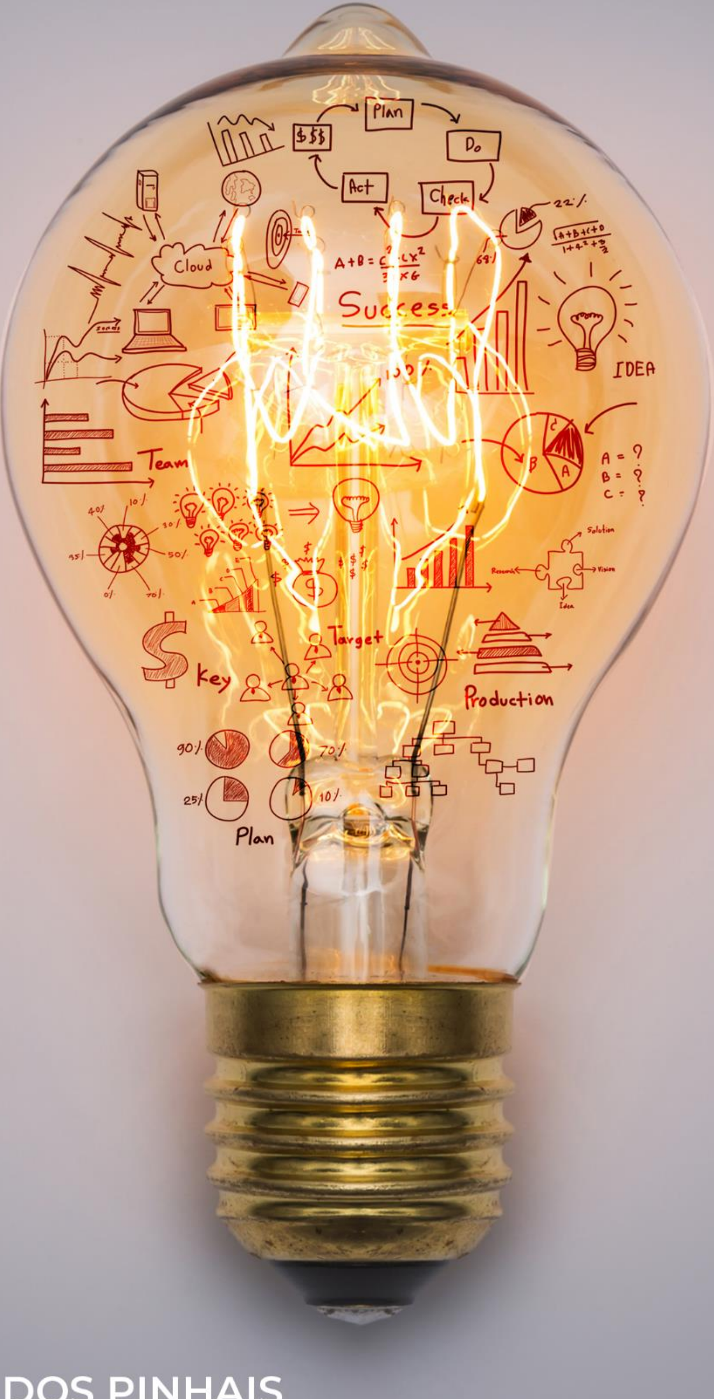
BARBARA BONFIM CATAPAN
HEAD ORGANIZER

INTERDISCIPLINARY PERSPECTIVES



SÃO JOSÉ DOS PINHAIS
BRAZILIAN JOURNALS PUBLICAÇÕES
DE PERIÓDICOS E EDITORA
2024

BrJ

Barbara Bonfim Catapan
Head Organizer

Interdisciplinary perspectives

Brazilian Journals Editora
2024

Diagramação: Editora
Edição de Arte: Editora
Revisão: Os Autores

Dados Internacionais de Catalogação na Publicação (CIP)

C357i Catapan, Barbara Luzia Sartor Bonfim

Interdisciplinary perspectives / Editora, Brazilian Journals. São José dos Pinhais: Barbara Luzia Sartor Bonfim Catapan, 2024..

Formato: PDF
Requisitos de sistema: Adobe Acrobat Reader
Modo de acesso: World Wide Web
Inclui: Bibliografia
ISBN: 978-65-6016-079-8

1. Multidisciplinar. 2. Aprendizado.
I. Catapan, Barbara Luzia Sartor Bonfim. II. Título.

Brazilian Journals Editora
São José dos Pinhais – Paraná – Brasil
www.brazilianjournals.com.br
editora@brazilianjournals.com.br

Ano 2024

PRESENTATION

Reading Interdisciplinary Perspectives is an invitation to reflect and delve deeper into the realm of human knowledge, encompassing diverse and complementary approaches. This book presents a journey intended for teachers, students, and professionals across all fields, fostering a broad and integrated view of the phenomena around us.

Interdisciplinarity, as an approach to study, enables us to challenge the boundaries of disciplines and explore new connections between seemingly distant areas of knowledge. Interdisciplinary Perspectives becomes an essential tool for those who seek not only to understand but also to question and expand the scope of their own knowledge.

For teachers, this work offers not only content but also new perspectives on how to integrate different areas of knowledge into their pedagogical practices. For students, it is an opportunity to explore the vast field of knowledge in a dynamic and interconnected way, encouraging the development of a critical and creative mind. Finally, for professionals, this book presents new ways to approach complex problems, which are crucial for the constant evolution of practices in any field.

Through the pages of Interdisciplinary Perspectives, it is hoped that the reader will develop a renewed appreciation for the richness of exchanges between areas of knowledge, building a solid foundation for the application of innovative and creative solutions in their respective fields of action.

SUMMARY

CHAPTER 1

A HYBRID ALGORITHM FOR CIRCLE DETECTION IN BINARIZED IMAGES

Emerson Vitor Castelani
Departamento de Matemática, Universidade Estadual de Maringá (UEM)
Address: Colombo Avenue 5790, Maringá, Paraná, Brazil.
E-mail: evcastelani@uem.br

Jesus M. Camargo
Universidade Estadual do Oeste do Paraná (UNIOESTE)
Address: R. Universitária, 1619, Cascavel, Paraná, Brazil
E-mail: marco.s.camargo@hotmail.com

Wesley V. I. Shirabayashi
Departamento de Matemática, Universidade Estadual de Maringá (UEM)
Address: Colombo Avenue 5790, Maringá, Paraná, Brazil.
E-mail: wvishirabayashi@uem.br

Jair da Silva
Universidade Federal do Paraná (UFPR), Campus Avançado de Jandaia do Sul
Address: R. Dr. João Maximiano, Centro, Jandaia do Sul, Paraná, Brazil
E-mail: jairmt@gmail.com

André L. M. Martinez
Departamento de Matemática, Universidade Tecnológica Federal do Paraná (UTFPR)
Address: Alberto Carazzai Avenue, 1640, Cornélio Procópio, Paraná, Brazil
E-mail: andrelmmartinez@yahoo.com.br

Eduardo de Amorim Neves
Departamento de Matemática, Universidade Estadual de Maringá (UEM)
Address: Colombo Avenue 5790, Maringá, Paraná, Brazil.
E-mail: eaneves@uem.br

ABSTRACT: In this work a hybrid algorithm that combines the Hough Transform and the Gauss-Newton method is presented for detecting circle in images. The basic idea of the new proposal is, for each change in discretized radius for the Hough Transform, to solve a convenient continuous optimization problem using the Gauss-Newton method. The goal of the optimization method inserted is to accelerate the obtaining of the correct radius and, at the same time, to apply a correction in the center provided by the discretization. Examples and comparisons with the classical version of the Hough Transform are explored to show the potential of the new proposal.

KEYWORDS: circle detection, gauss-newton method, hough transform, hybrid method.

RESUMO: Neste trabalho, é apresentado um algoritmo híbrido que combina a Transformada de Hough e o método de Gauss-Newton para detectar círculos em imagens. A ideia básica da nova proposta é, para cada mudança no raio discretizado para a Transformada de Hough, resolver um problema de otimização contínua conveniente usando o método de Gauss-Newton. O objetivo do método de otimização inserido é acelerar a obtenção do raio correto e, ao mesmo tempo, aplicar uma correção no centro fornecida pela discretização. Exemplos e comparações com a versão clássica da Transformada de Hough são explorados para mostrar o potencial da nova proposta.

PALAVRAS-CHAVE: detecção de círculos, método de gauss-newton, transformada de hough, método híbrido.

1. INTRODUCTION

Problems of detection of geometric shapes in binarized images are widely studied in the field of computer vision with important applications. Given a binarized image of size $r \times s$, consider that the amount of points in that image is given by t. In this case, you can associate it with a set of points in the plane given by $Ib = \{(a_i, b_i) \in N \times N, i = 1, \dots, t\}$. Suppose you want to detect a geometric shape represented by $\phi(x_1, x_2, \dots, x_n, a, b) = 0$, where $\phi: R^n \times R^2 \to R$. A priori, it ϕcan describe any curve: a straight line, a circle, an ellipse, a parabola, etc. In this work, a proposal is presented for the case in which it $\phi$is a circle. Formally, one intends to find a subset of points $FIb \subset Ib$and parameters x_1, x_2, x_3such that $\forall(a, b) \in FIb$, one has $\phi(x_1, x_2, x_3, a, b) = (a - x_1)^2 + (b - x_2)^2 - x_3^2 \approx 0$.

Of course, the problem formulated is a discrete problem and in this context, one of the most renowned forms of resolution is the Hough Transform proposed in [1] and its variations [2,3,4,5,6]. Ideas based on the Hough Transform depend on the discretization of the search space of the parameters to be determined and an accumulation matrix (which, in our context, will be called a tensor due to its three-dimensional structure) is generated and associated with this discretization.

Other resolution techniques through continuous models combined with random point selection are also explored successfully. In this case, the RANSAC method proposed in [7] is highlighted. Another interesting approach is dealt with in [8], where the authors explore a continuous optimization model based on ordered functions of type OVO [9] and propose a new model to solve the problem of detection of geometric shapes in images using an adaptation of the Gauss-Newton method (classic for solving problems of least squares). In this case, the authors explore some comparisons of the new approach with the RANSAC method and the Hough Transform, obtaining very competitive results. In this text, such an algorithm will be called GNFO (Gauss-Newton for Ordered Functions).

The Hough Transform is a very robust method that is even present in famous computer vision libraries, such as OpenCV (www.opencv.org). However, it is well known in the literature that when the number of parameters to be determined grows, the Hough Transform greatly increases the demand for memory and may, due to the discretization scheme, become time-consuming to perform detection.

On the other hand, the GNFO method, which is an iterative method, when converging uses few iterations and therefore consumes few memory and processing resources. However, it is quite common that depending on the choice of initial approximation, the GNFO method stops at stationary points of the optimization model that are local minima, which in general does not provide correct detection. Thus, multi-start strategies are needed to obtain the overall minimizer of the model and thus the correct detection.

In order to speed up the detection of circles by the Hough Transform and correct the demand for multi-start strategies of the GNFO method, a hybrid algorithm is proposed in this work such that for each prefixed radius, the possible center (most voted at that time) obtained by an intermediate step of the Hough Transform is used as the starting point of the GNFO method which, in this case, will try to obtain the correct radius and center. In the worst case, the optimization method does not perform corrections and at the end of the process, one obtains the same solution as the Hough Transform. However, if the optimization algorithm is successful, a reduction in processing time and memory consumption is expected in relation to the Hough Transform presented in this paper.

2. MATERIALS AND METHODS

In this section, the main concepts and models for the new algorithm proposal will be clarified.

2.1 THE LOVO PROBLEM

The Low Order Value Optimization (LOVO) problem is a class of optimization problems derived from the Order Value Optimization (OVO) problem. Such problems were introduced in [10] and since then variations and applications have been explored in several directions, a proper revision using [11,12,13] and [14] is recommended. In a general context, the LOVO problem can be described as follows. Consider $F_i: R^n \to R, i = 1, \dots, m$,continuous functions with continuous derivatives. For each one $x \in R^n$the set of images can be arranged in an increasing order $F_i(x), i = 1, \dots, m$. Thus, if the indices $\{i_k(x), k = 1, \dots, m\}$ denote this ordering, one has:

$F_{i_1(x)}(x) \leq F_{2(x)}(x) \leq \cdots \leq F_{i_m(x)}(x).$ (1) (2)

Given $p \leq m$, the LOVO function is defined by

$$S_p(x) = \sum_{i=1}^{p} F_{i_k(x)}(x) \quad (1)$$

and the LOVO problem is to minimize this function, ie,

$$min S_p(x).$$

As highlighted in [12], the LOVO problem is a generalization of the method of least squares. In fact, denote by $\{(t_i, y_i), i = 1, \ldots, m\}$ set of observations whose model of adjustment is described by $\varphi(x,t)$ where $x \in R^n$ is the vector of parameters to be determined. Thus, if $F_i(x) = (\varphi(x,t_i) - y_i)^2, i = 1, \ldots, m$ we denote deviations from φ in $t_i$relation to y_i, we have for $p = m$that the model that LOVO generated, in this case, will be equal to the problem of least squares. On the other hand, if $p < m$ then the generated problem will provide a model such that the $m - p$ worst deviations will be discarded, i.e. in this case the LOVO problem discards the influence of possible "outliers".

2.2 RESOLUTION OF THE LOVO PROBLEM IN THE CONTEXT OF CIRCLE DETECTION

Consider, again, a binarized image with t dots. For each one i =1, …, t one can define $F_i(x_1, x_2, x_3) = ((a_i - x_1)^2 + (b_i - x_2)^2 - x_3^2)^2$ where ai,bi∈Ib. Let p<t the number of points represent a circle detection. Thus one must find values x1, x2, x3 and a subset Δ⊂{1,2,…,t} of cardinality p, such that Fix1,x2,x3≈0 for the whole i∈Δ. Otherwise , it is desired to find the global minimizer of the following LOVO problem:

$$\min_{x \in R^3} S_p(x) = \min_{x \in R^3} \sum_{k=1}^{p} \left(\left(a_{i_k(x)} - x_1\right)^2 + \left(b_{i_k(x)} - x_2\right)^2 - x_3^2 \right)^2 \quad (3)$$

The Gauss-Newton method is widely used in solving nonlinear least squares problems. The main advantage of this type of method is the omission of second-order derivative calculations, while maintaining order of quadratic convergence, it is recommended [15] and [16]. Of course, the above problem is continuous, but the objective function may not be derivable. Thus, the application of the Gauss-Newton method is not immediate because we do not have gradients available at all points. However, subgradients generated by sorting can be used. In this sense, the authors of [8] proposed, denoting

$$r_i(x_1, x_2, x_3) = (a_i - x_1)^2 + (b_i - x_2)^2 - x_3^2, i = 1, \dots, t$$

and

$$F_i(x_1, x_2, x_3) = r_i(x_1, x_2, x_3)^2, i = 1, \dots, t,$$

the following algorithm.

Algorithm 1 (GNFO for circle detection)

Be $x_0 \in R^3$ an initial approximation for the parameters of the desired curve and $\varepsilon > 0$ a given tolerance. Let $x_k \in R^3$ the approximation of parameters in iteration *k*. Then, the new approximation x_{k+1} is obtained as follows:

1. Calculate $\nabla r(x_k) = \left(\nabla r_{i_1(x_k)}(x_k), \dots, \nabla r_{i_p(x_k)}(x_k)\right)$
2. Solve the Gauss-Newton system $\nabla r(x_k)\nabla r(x_k)^T d = -\nabla r(x_k) r(x_k)$.
3. Get α ∈ such that $S_p(x_k + \alpha d) \leq S_p(x_k) + \alpha d^T \nabla r(x_k) r(x_k)$.
4. $x_{k+1} = x_k + \alpha d, k = k + 1$
5. If $\|\nabla r(x_k) r(x_k)\| < \varepsilon$ stop and return x_k.

2.3 THE HOUGH TRANSFORM IN THE CONTEXT OF CIRCLE DETECTION

Although the formulation of the Hough Transform is simple, the strategy is widely used in practical situations (see [17]). A conceptual algorithm of this technique is described below.

Algorithm 2 (Hough transform for circle detection)

Consider $Ac = 0$ null tensor and $r_{min}, r_{max} \in N$ minimum and maximum values for the possible radius of the circle.

1. To $r = r_{min}, \dots, r_{max}$ do
2. To $(a, b) \in Ib$ do
3. To $\theta = 1°, 2°, \dots, 360°$ do
4. $i = a - rcos\theta$
5. $j = b - rsin\theta$
6. $Ac[i, j, r] = Ac[i, j, r] + 1$
7. End (To)
8. End (To)
9. End (To)
10. Determine i, j, k such that $max(Ac[:, :, :]) = Ac[i, j, k]$.
11. Return $x_1 = i, x_2 = j, x_3 = r$.

The tensor Ac corresponds to the voting system. The input of the tensor that had the highest value will provide the parameters sought. The conceptual algorithm allows for several modifications and some improvements. Although storage resources are expensive in the algorithm presented, their performance is comparable with other variations of the said method, it is recommended [18].

2.4 THE NEW ALGORITHM

The central idea of the new method is, with each change of radius from Algorithm 2, one has in the accumulation matrix (two-dimensional to fixed-radius) a good initial approximation to run Algorithm 1 (GNFO). Hence, the name of "Hybrid". Furthermore, as the optimization method tries to find a solution that may have different values (radius and center) from the initial approximation, we can decrease the discretized space of radius and angle. It is therefore clear that the intention of the new proposal is to accelerate the achievement of a circle by the Hough Transform. Denoting by Δ_r and Δ_θ subsets of $\{r_{min}, \dots, r_{max}\}$e $\{1°, 2°, \dots, 360°\}$, one has the following conceptual algorithm regarding the new strategy.

Algorithm 3 (Hybrid Method for Circle Detection)

Consider $Ac = 0$ a null tensor and Δ_re $\Delta_\theta$as defined above. Consider further, $tol > 0$ a tolerance and $v_{best} \in R^4$.

1. To $r \in \Delta_r$do
2. To $(a, b) \in Ib$do
3. To $\theta \in \Delta_\theta$do
4. $i = a - rcos\theta$
5. $j = b - rsin\theta$
6. $Ac[i, j, r] = Ac[i, j, r] + 1$
7. End (To)
8. End (To)
9. Determine i, jsuch that $max(Ac[:, :, r]) = Ac[i, j, r]$.
10. It $v_{best}[4] < Ac[i, j, r]$is
11. $v_{best}[1] = i, v_{best}[2] = j, v_{best}[3] = r$
12. $v_{best}[4] = Ac[i, j, r]$
13. End (If)
14. Apply Algorithm 1, getting $\underline{x}_1, \underline{x}_2, \underline{x}_3$. If $S_p(\underline{x}_1, \underline{x}_2, \underline{x}_3) < tol$, stop and return $x_1 = \underline{x}_1, x_2 = \underline{x}_2, x_3 = \underline{x}_3$.
15. End (To)
16. Return $x_1 = v_{best}[1], x_2 = v_{best}[2], x_3 = v_{best}[3]$.

The vector v_{best} introduced in Algorithm 3 is used to store the best solution obtained by the Hough Transform for a given radius. Thus, if the optimization method does not force the algorithm to stop, one can use it v_{best} to provide the solution without searching again for the tensor Ac. On the other hand, if at some stage of the variation of the radius, the method defined by Algorithm 1 finds a "good" solution (in the sense set out in Algorithm 3 step 14) we stop the execution of the method. Of course, an adaptation of the strategy defined can provide multiple circles. However, in this case, the process cannot be restricted to a single call to Algorithm 1 by radius and also, the phase of the Hough Transform should not store only one (best) solution. In this sense, the acceleration phase does not compromise robustness, but generates more processing for the Hough Transform method, as Algorithm 3 would not stop by solving the optimization problem without analyzing other possible ones. For this reason, for the detection of multiple circles, it is more advantageous to use a ‘multistart’ strategy in the sense proposed in [11]. Therefore, the study in hand is limited to the case of detection of only one geometric shape per image.

Finally, it should be noted that the tolerance ε of Algorithm 1 is different from the *tol* tolerance of Algorithm 3. In Algorithm 1 this tolerance measures stationarity, which is an inherent value of the method while in Algorithm 3, *tol* qualifies the solution that is a measure related to the problem. Consequently, for practical reasons, it is suggested to consider $\varepsilon = 10^{-4}$ and $tol = 15p$ where p is the estimated number of points for a possible circle.

3. RESULTS AND DISCUSSION

Considering real problems, we considered 10 examples of images to test the implementations and compare them. The outlines of the images were extracted using an implementation of the Sobel algorithm in Julia (www.julialang.org), version 0.6. Implementations were tested on an Intel Pentium (R) CPU G3240 computer, 3.10GHz, 4GB of RAM and Ubuntu-Mate operating system 16.04. The size of each image (TI- in pixels) and the number of points resulting from binarization (QP) are highlighted in Table I.

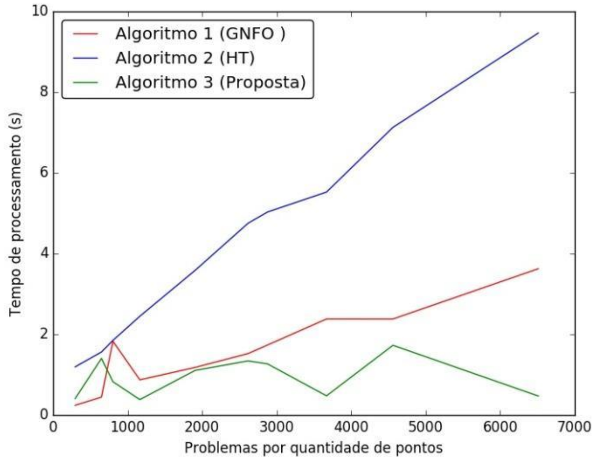
Table I. Comparison of algorithms in real images.

Problem	IT	QP
Coins	240x160	803
Alo	252x154	1,164
Rand	756x756	2,671
Rand 2	756x756	2,878
Sunflower	204x204	296
Dishes	600x600	4,561
Football Ball	225x225	649
Dishes 2	292x292	3,669
Vinyl	600x600	6,511
Vinyl 2	500x233	1,907

Source: Prepared by the author.

In Figure 1 and 2, we highlight the behavior regarding processing time and memory consumption, respectively.

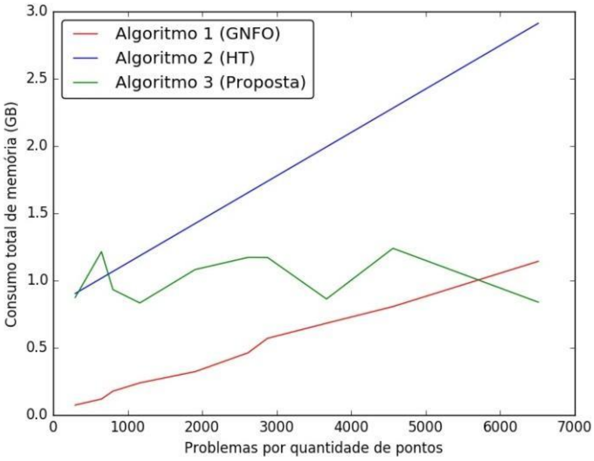
Figure 1: Comparison between the algorithms presented, considering the processing time to find the solution or meet the halting criteria.



Source: Prepared by the author.

Illustrations of the results obtained by the three algorithms are shown in Fig. 3, 4 and 5, considering the smallest images tested.

Figure 2: Comparison between the algorithms presented, considering the memory consumption allocated to find the solution or meet the stop criteria.

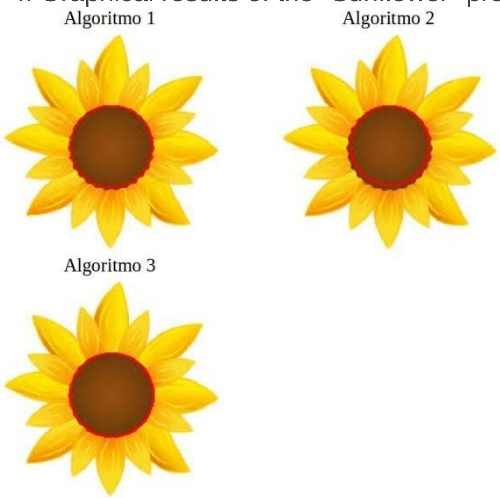


Source: Prepared by the author.

Figure 3: Graphical results of the "Currencies" problem.

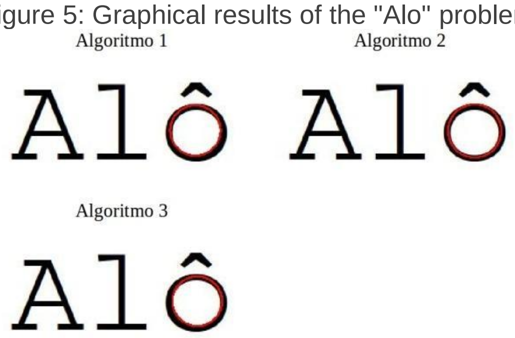
Source: https://www.bcb.gov.br/cedulasemoedas/moedascomemorativas (access 10/04/2019)

Figure 4: Graphical results of the "Sunflower" problem.



Source: www.flaticon/free-icons/agriculture_77871 (accessed 17/07/2017).

Figure 5: Graphical results of the "Alo" problem.



Source: Prepared by the author.

Naturally, as can be seen in Fig. 3, the circle obtained may be different, depending on the method used. Also, given the characteristic of the optimization method in generating sequences whose limit point is attracted to some local minimizer, it has been found that some detections of Algorithm 1 were not adequate.

Cases like this can be minimized with modifications in Algorithm 1 to filter out the solutions (decide if the dots are on a circle) and remove dots from the image (preventing the algorithm from stopping again at these points). Such techniques generate more processing and are usually associated with the detection of multiple circles, which is not the focus of the work at hand. Examples where Algorithm 1 was not effective in detection (stopped at weak stationary points) were: "Rand", "Rand 2", "Dishes", "Dishes 2" and "Vinyl 2". The other methods did not exhibit this type of behavior and their detections were quite satisfactory.

To complement our tests, artificial instances were generated to simulate noise and to simulate clusters of points. In total, 32 instances were created to simulate noise where a set of random points was created for each image with different density as well as the disturbance of a certain amount of points of the circumference to be detected. Also, 14 images were generated with clusters of points created in random positions with different densities in each image. All algorithms, images and problems presented in the present work can be obtained at www.github.com/evcastelani/curve_detection. Table II expresses the results obtained by each algorithm considering the processing time where the shortest time is highlighted. In this table, QP denotes the number of points in the instance. All issues correspond to 300x300 images. In the problems Noise 1-21 and Cluster 1-10, p=25 was taken to run Algorithms 1 and 3. In the other problems p=50.

Table II. Comparison of algorithms in artificial instances.

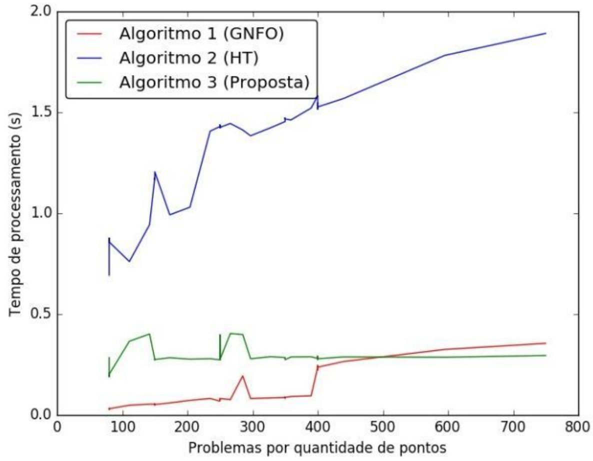
Problem	**QP**	**Algorithm 1**	**Algorithm 2**	**Algorithm 3**
		TP	**TP**	**TP**
Noise 1	80	**0,033994298**	0.692724545	0.284152183
Noise 2	80	**0,032686641**	0,876928274	0,189274264
Noise 3	80	**0,033025564**	0,866679232	0,190590441
Noise 4	80	**0,032743462**	0,859649774	0,190135006
Noise 5	80	**0,031729515**	0.857813018	0.202263106
Noise 6	150	**0,047295885**	0.95829998	0.191866633
Noise 7	150	**0,045165014**	0.957342206	0,190437747
Noise 8	150	**0,047350592**	0.959125104	0,191063576
Noise 9	150	**0,047484061**	0.961616089	0,189686458
Noise 10	150	**0,046880287**	0.962727991	0.189204538
Noise 11	250	**0,069554718**	1,192934146	0,190718773

Problem	QP	Algorithm 1	Algorithm 2	Algorithm 3
		TP	**TP**	**TP**
Noise 12	250	**0,073868396**	1.189933273	0,307841729
Noise 13	250	**0,075379789**	1.189940887	0.307598643
Noise 14	250	**0,069178156**	1.194398933	0.191189908
Noise 15	250	**0,075507888**	1,190074744	0,189430093
Noise 16	250	**0,072767003**	1.189649604	0,193077406
Noise 17	350	**0,086705947**	1.332915823	0.196435193
Noise 18	350	**0,087630827**	1.333618466	0,196690409
Noise 19	350	**0,084583653**	1.324788128	0,190822175
Noise 20	350	**0,085760567**	1.329376697	0,190789442
Noise 21	350	**0,085621345**	1.326425868	0,190587715
Noise 22	400	0.258305394	1.405541239	**0.189999493**
Noise 23	400	0,255760117	1.380384221	**0,190059874**
Noise 24	400	0,262686123	1.389552479	**0.227020642**
Noise 25	400	0.245137196	1.389510871	**0.213939797**
Noise 26	400	0,26397979	1.396549909	**0,19115216**
Noise 27	400	0,263838751	1.389722592	**0.198141937**
Noise 28	400	0.26781753	1.386725955	**0,19219469**
Noise 29	400	0.253201376	1.380371945	**0,190345611**
Noise 30	400	0,240427634	1.39309123	**0,190501181**
Noise 31	400	0,247652959	1.406869439	**0,189740916**
Noise 32	400	0.254219941	1.394191869	**0,192144962**
Cluster 1	111	**0,044472499**	0,760775074	0,188364476
Cluster 2	142	**0,050593374**	0.943259095	0.301219455
Cluster 3	173	**0,060259212**	0.992278528	0,188844081
Cluster 4	204	**0,064651725**	1.029995465	0.189988383
Cluster 5	235	**0,074189536**	1.161974478	0,196830461
Cluster 6	266	**0,0738254**	1.198566789	0.298973266
Cluster 7	297	**0,078751451**	1.228774371	0,190044837
Cluster 8	328	**0,08322501**	1.278012639	0.196824963
Cluster 9	359	**0,085880044**	1.313788558	0.197208898
Cluster 10	390	**0,103976215**	1.377751013	0,197902469
Cluster 11	285	**0,19232339**	1.361256565	0,233603196
Cluster 12	440	0.260241992	1 4498 75 81	**0.213521444**
Cluster 13	595	0.32018646	1 652 91 37 92	**0,218407642**
Cluster 14	750	0,373861029	1 841 121 908	**0.221032341**

Source: Prepared by the author.

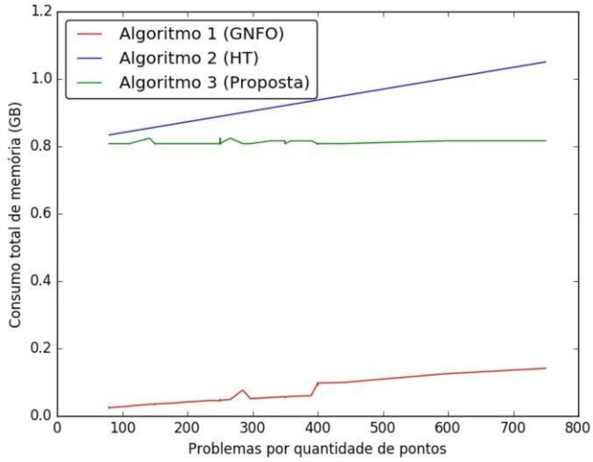
For an overview of the performance of the presented algorithms, we illustrate in Fig. 7, the behavior expressed in Table II as a function of the number of points (QP). Also, to illustrate the memory consumption, we highlight in Fig. 8, the behavior of the algorithms in this question, also, as a function of the number of points (QP).

Figure 7: Algorithm performance in artificial instances considering processing time versus number of points (QP).



Source: Prepared by the author.

Figure 8: Algorithm performance in artificial instances considering memory consumption versus number of points (QP).



Source: Prepared by the author.

4. CONCLUSIONS

In this work a new algorithm was presented that corresponds to a hybrid strategy based on the Hough Transform and the Gauss-Newton method to solve a convenient optimization problem using ordered functions. The new proposal showed promising practical results, as detections were obtained, in general, with shorter processing time than strategies based purely on the Hough Transform or based purely on the Gauss-Newton method (GNFO) when considering real problems. In addition, memory consumption was moderate in all tested examples, i.e., although the new approach does not have as low memory consumption as the GNFO method, it shows substantial improvement over the Hough Transform.

When considering artificial instances, all methods found the solution sought, and in this case, the method that performed best in processing time was the GNFO method whose performance in 32 problems was superior to the other methods. In another 14 problems, the hybrid method was superior in this respect. It should be noted, however, that the performance of the hybrid method improves as the number of points of the instances increases. Indeed, when the number of points increases in the generated instances, the number of local minima increases, and therefore the starting points obtained by the Hough transform steps become good starting points (in the sense of approximating the global minimum) for Algorithm 3 step 14.

Another advantage that should be highlighted in relation to the new proposal is that, in the worst case, the robustness is equivalent to the Hough transform. This robustness is achieved due to the hybrid nature of the new algorithm. Therefore, the practical potential of the work is relevant. In fact, as highlighted earlier, the new proposal has adequately found a circle in all the real problems tested while the GNFO method has not been adequately detected in some real examples.

On the other hand, it is important to point out that the new proposal is focused on detecting only one circle. Without a proper modification of the proposed algorithm, the performance for this purpose is compromised as observed in the previous section. Considering the detection of multiple geometric forms, the results presented in [8] suggest that the GNFO method has great potential for this purpose. In this sense, it is intended as future work:

- Propose a new hybrid method for the detection of multiple geometric shapes;
- Increase the variation of geometric shapes;
- Establish a filter scheme to refine the solutions obtained.

REFERENCES

[1] R. O. Duda; P. E. Hart, "Use of Hough Transform to detect lines and curves in pictures". *Communications of Association Computing Machinery*, 15, 1972.

[2] (A) E. Cowart; W. E. Snyder; W. H. Ruedger; ‘The detection of unresolved targets using the Hough Transform’. *Computer Vision, Graphics and Image Processing*, 21, pp. 222-238, 1983.

[3] L.M. Murphy. "Linear feature detection and enhancement in noisy images via the Radon Transform." *Pattern Recognition Letters,* 4, p. 279-284, 1986.

[4] (J) Illingworth.;J. Kittler. A survey of The Hough Transform. *Computer Vision, Graphics and Image Processing,* 44, pp. 87-116, 1988.

[5] C. Ho; L. H. Chen. A high speed algorithm for line detection. *Pattern Recognition Letters*, 17. P.467-473, 1996.

[6] O. Chutatape; L. Guo. "A modified Hough Transform for line detection and its performance", *Patern Recognition*, 32,p.181-192,1999.

[7] M. A. Fischler; R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography". *Communications of ACM*, 24, pp. 381-395, 1981.

[8] R. Andreani; G. Cesar; J. M. Martínez; P. J. S. Silva., ‘Efficient cruve detection using Gauss-Newton method with applications in agriculture’. *1st International Workshop on Computer Vision Applications for Developing Regions in Conjunction with ICCV 2007,*pp. 1-13, 2007.

[9] R. Andreani; C. Dunder; J. M. Martínez, ‘Nonlinear programming reformulation of the order value optimization problem’. *Mathematical Methods of Operations Research*, 61, pp. 365-384, 2005.

[10] R. Andreani; C. Dunder; J. M. Martínez, ‘Order Value Optimization: Formulation and solution by means of a primal Cauchy Method’. *Mathematical Methods of Operations Research*, 58, pp. 387-399, 2003.

[11] R. Andreani; J. M. Martínez; L. Martínez; F. Yano, ‘Continuous optimization methods for structure alignments’. *Mathematical Programming,* 112, p. 93-124,2008.

[12] R. Andreani; J. M. Martínez; L. Martínez; F. Yano, ‘Low Order Value Optimization and applications’. *Journal of Global Optimization*, 43, pp. 1-22, 2009.

[13] E. P. Carvalho; F. Pisnitchenko; N. Mezzomo; S. R. S. Ferreira; J. M. Martínez; J. Martínez, "Low Order Value Multiple Fitting for supercritical fluid extraction models". *Computers and Chemical Engineering*, 40, p. 148-156,2012.

[14] (J) M. Martínez, ‘Generalized order value optimization’, *Top (Madrid)*, 20, pp. 75-98, 2012.

[15] (D) Bertsekas, *Nonlinear programming.* Athena Scientific Belmont, 1999.

[16] H. Hartley, "The modified Gauss-Newton Method for fitting of nonlinear regression function by least squares". *Technometrics*, 3, pp. 269-280,1961.

[17] G. J. Bergues; L. Canali; C. Schurrer; A. G. Flesia, "Sub-Pixel Gray-Scale Hough Transform for and Electronic Visual Interface", *IEEE Latin America Transactions,* 13, p. 3135-3141,2015.

[18] H. K. Yuen; J. Princen; J. Illingworth, ‘Comparative Study of Hough Transform methods for circle detection’. *Image and Vision Computing*, 8, pp. 71-77, 1990.

Agência Brasileira ISBN
ISBN: 978-65-6016-079-8