Electronic Letters on Computer Vision and Image Analysis 16(1):33-53, 2017

AN ANT COLONY BASED MODEL TO OPTIMIZE PARAMETERS IN INDUSTRIAL VISION

Benchikhi loubna* and Sadgal Mohamed* and Elfazziki Aziz* and Mansouri Fatimaezzahra*

* *Department of computer science, Computer Systems Engineering Laboratory , Bd. Prince Moulay Abdellah, B.P. 2390, 46000, Marrakesh, Morocco*

l.benchikhi@uca.ma, sadgal@uca.ma, elfazziki@uca.ma, fatimaezzahra.mansouri@ced.uca.ma.

Received 08 th Jul 2016; accepted 09 th May 2017

Abstract

Industrial vision constitutes an efficient way to resolve quality control problems. It proposes a wide variety of relevant operators to accomplish controlling tasks in vision systems. However, the installation of these systems awaits for a precise parameter tuning, which remains a very difficult exercise. The manual parameter adjustment can take a lot of time, if precision is expected, by revising many operators. In order to save time and get more precision, a solution is to automate this task by using optimization approaches (mathematical models, population models, learning models...). This paper proposes an Ant Colony Optimization (ACO) based model. The process considers each ant as a potential solution, and then by an interacting mechanism, ants converge to the optimal solution. The proposed model is illustrated by some image processing applications giving very promising results. Compared to other approaches, the proposed one is very hopeful.

Key Words: Image processing, Industrial vision, Ant colony optimization, Quality control

1 INTRODUCTION

Processing an image involves its transformation into results, which may be a new image or some features. These last are resulting from the application of different operators, where each one has a set of adjustable parameters. The choice of operators to apply and their free parameters' adjustment affect mainly the quality of results.

The lack of a general rule that guides the user in his choices, forces him to go through a process of manual parameter adjustment by a trial-and-error process, which is the conventional approach to find suitable values: Users provide parameter values, initialize and wait for algorithms to execute. The output is inspected, parameter values are changed, and the process repeated until satisfactory output is produced.

In particular, application of image processing algorithms for quality control requires users to find the parameter values to correctly detect imperfections. In order to help users achieving vision tasks, several solutions have been proposed such as programming by graphs, Ariane (Visual programming interface for operators' chains) based on Pandore (a normalized library of image processing operators) composed of a set

Correspondence to: loubnabenchikhi@gmail.com

Recommended for acceptance by Angel Sappa
http://dx.doi.org/10.5565/rev/elcvia.957
ELCVIA ISSN: 1577-5097
Published by Computer Vision Center / Universitat Autonoma de Barcelona, Barcelona, Spain

of programs directly executable on image files. However, these works remain semi-manual since it requires user intervention.

In other terms, this problem can be presented in two issues:

- How to choose operators and fix their parameters' values to have the best result for a particular vision task?
- Often, it is the user who judges final results and it is difficult to give a formal aspect of the problem definition. In this case, how to automate a process with no formal definition?

Some authors searched to completely automate the process of choosing operators and adjusting their parameters by proposing methods such as numerical optimization, applying mathematical or statistical techniques to minimize, or maximize, an objective function defined over parameter space [1]. All the proposed techniques have not been widely adopted. This can be partly awarded to few application examples in the real image analysis. Visually, it is an NP-complex problem and our objective is to use Artificial Intelligence techniques for solving such problems. The contribution of this paper is an answer to the two questions above, by proposing a novel model using Ant Colony Optimization (ACO). Optimal parameter values for applications in vision systems are found. It is shown that using ACO allows achieving higher quality results. The approach is applied on real image examples in quality control tasks and the outputs are compared to the references.

This study is a part of an assembly of models [2,3] designed to automate the parameters setting process in image processing. The main objective is to provide a new model based on the ACO approach beside those already proposed: firstly, to compare their performances. Secondly, to get around the difficulties of expressing this problem in a mathematical form accessible by conventional algorithms. The remainder of this paper is organized as follows: Section 2 provides an overview of studies done in this context. In section 3, a basic idea of a vision task is explained. Section 4 explains the novel ACO model for parameter optimization problem. The established ACO model is conducted on several real quality control examples together with a discussion on experimental results in Section 5. Section 6, devoted to draw a general conclusion and a discussion of different refinement possibilities is done.

2 RELATED WORK

In mathematics and computer science, an optimization problem is to find the best solution among all feasible ones. An optimization problem with discrete variables is known as a combinatorial one, where an object such as an integer, a graph or a combination is looked for, from a finite set.

Parameter adjustment in image processing can be considered as a combinatorial optimization problem, while we are searching for the best parameter values' combination over all possible ones.

In real applications, many methods were used to find the best parameters' values such as experimental designs [4] based on statistics. On entry, this procedure uses a matrix of experiences (essay-values) and it tries to generalize by a linear function: response-values. The work presents a variant that detects parameters influence and reduces the number of tries compared to direct trial-and-error method.

A different technique was introduced [5], using interactive visualization to develop novel histopathology image segmentation software, which illustrates its potential usefulness for parameter optimization purposes.

In recent years, parameters optimization in image processing is supported by artificial intelligence techniques [6], such as multi-agent architecture. I.Qaffo and al. [3] proposed an automatic method based on reinforcement learning for object recognition, using two agents: User Agent (UA) and Parameter Agent (PA). The UA gives necessary information to the system, as the combination of applicable operators, the set of adjustable parameters for each operator, and a values' range for each parameter. The PA uses reinforcement learning to assign the optimal values for each parameter in order to extract the object of interest from an image.

One of the most used methods for parameter optimization in image processing is population-based heuristics [7,8]. Genetic algorithms [9] are widely used for this purpose. We proposed an optimization method based on particle swarm optimization [2] to find the best values of free algorithm parameters used in image processing. Further and based on this previous work, we reconsider a model that finds the best combination of operators, besides the best values of their parameters based also on a population heuristic [10]. This method proves its feasibility and efficiency.

On the other hand, ant colony optimization (ACO) algorithm was first introduced by A.Colorni and al. [11] [12] in the nineteens, as a novel nature-inspired method from the ants' behavior in their cohabitation in colonies. It was introduced as a solution of combinatorial optimization problems. From then on, many researchers have applied ACO to different optimization problems like vehicle routing [13], transportation planning problems [14], job shop scheduling [15], continuous optimization problems [26], software structural testing [17]. It was used also in image processing [18] for image segmentation [19] [20], image thresholding [21], image edge detection [22] [23], face recognition [24], structural topology optimization [25], optimization of support vector machines parameters' [26], image watermarking algorithm scaling factors' optimization [27] and image feature selection [28]. All these researches have proved the ACO efficiency as a new tool to obtain satisfying optimization results. This study adopts ACO to present a model to optimize vision systems operators' and parameters' values.

3 VISION TASK

3.1 Global definition

To accomplish a vision task, it is necessary to go through multiple tasks. It can be a set of other tasks or an elementary task composed of different phases of treatment (Figure 1). Each one contains a set of operators, and each operator has many parameters.

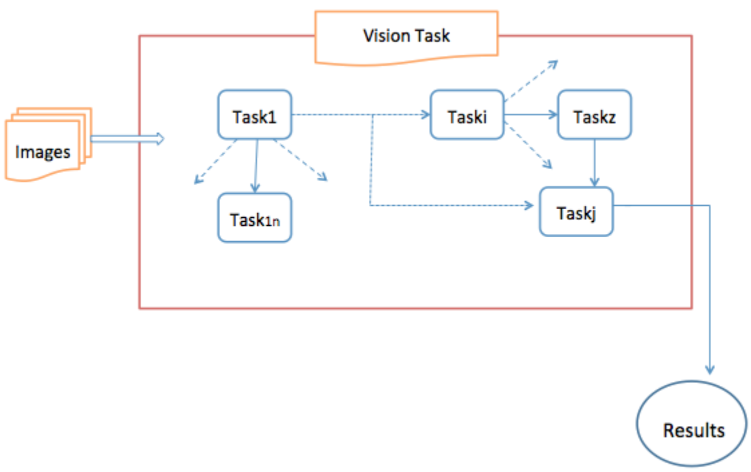


Figure 1. General process to implement computer vision systems

Users find themselves facing a tedious job, choosing the best operators to apply and adjusting their parameters. The following approach proposes a method helping users to address the deficiencies of previous ones. This approach relies on two axes:

- First: Making a set of operators available for each phase of treatment and fixing their parameters.
- Second: The ACO optimization model Application, to detect the best combination of operators, and their optimal parameters.

3.2 Operators in a vision task

A vision task is a succession of several phases. Each phase has a set of possible operators.

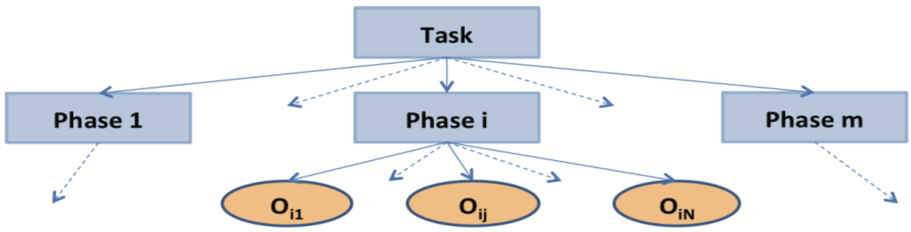


Figure 2. Vision task possible operators for each phase.

Each phase i may have one or more possible operators (O_{i1},…,O_{iN}). To accomplish a vision task, we must combine operators from different phases. So, several combinations of these operators: C_k, k=1… n. could be considered to find the best task result. Each C_k can achieve the considered task, but the output is qualitatively different. Let's consider a task made of several phases. Since each phase concocts a set of feasible operators, n different operator's combination C_k are built: $C_k = (O_1, O_2, ..., O_m)$ k = 1….n.

An operator (O_i) requires fixing some parameters; a range of possible values for each parameter is given out (figure 3).

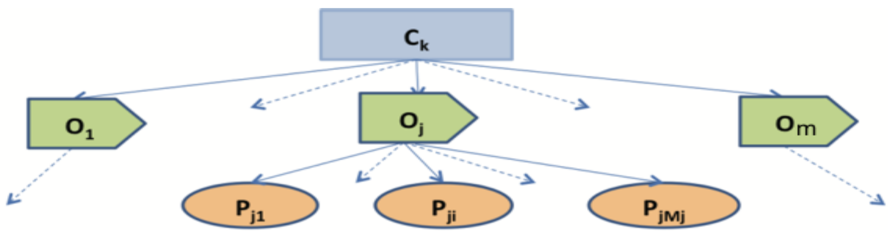


Figure 3. Operator's combination and their parameters.

4 The proposed approach for parameters' optimization

4.1 The Optimization Model

4.1.1 Definition of the Objective function

Mathematically the problem can be defined considering a model P = (D, w, Err) where:

- D is the solution space, defined over a set of variables, and w a set of constraints among the variables.
- An objective function Err: $D \rightarrow R^+$ to be minimized.

A solution $d \in D$ is a complete assignment in which each variable has an assigned value; d must satisfy all the w constraints. A feasible solution d* is a global optimum if: $Err(d^*) < Err(d)\ \forall d \in D$.

Back to our problem, an operators' combination C_k and an input image I_{input} are considered. The application of C_k provides an output result R_k (image or features):

$$R_k = C_k(I_{input}) \quad (1)$$

To evaluate the result, an objective function is necessary to compare qualitatively the outputs (R_k), a reference led by an expert is used as a ground truth R_r. The rating calculated by this objective function represents the error between the two images. The objective function, called also fitness function depends mainly on the vision task

$$Err_k(C_k) = fitness(R_k, R_r) \qquad (2)$$

For sure optimal parameters' values give the best output. Consequently, this corresponds to the smallest error (Err), and then the operators' combination to adopt is the one corresponding to the minimal error:

$$Err = Min_{k=1...n}(Err_k(C_k)) \qquad (3)$$

4.1.2 Space of Parameter values

Let's consider an operator's combination applied to the input image I_{input}, based on figure.3 and equation (1) we note:

$$I_{output} = (O_1, O_2, \ldots, O_N)(I_{input}) \ (4)$$

To simplify, we consider a combination as a sequence of N operators where each operator is applied over the output of its predecessor:

$$I_j = O_j (I_{j-1}) \quad j = 1...N, I_{input} = I_0, I_{output} = I_N = R_N \ (5)$$

Considering that each operator O_j has M_j parameters, and the $M = \sum_{j=1}^{N} M_j$ is the number of all parameters. Then the error is simply function of parameters denoted by:

$$Err (P_1, P_2, ..., P_M) = fitness (R_N , R_r) \ (6)$$

Where P_i, is a parameter taking values in a domain D_i. Then, we consider the Cartesian product $D = D_1 * D_2 * ... * D_M$, and we call a path of parameters' values each M-uplet: $(v_1, v_2,...,v_M) \in D$ where v_i is a value of parameter p_i. Using these notations, the problem is to find a path that minimizes the error function (Err) over the domain D. So, the solution is an M-uplet $(v_1^*, v_2^*, ..., v_M^*) \in D$, such as $Err (v_1^*, v_2^*,..., v_M^*)$ is minimal.

The objective function is not expressed directly with parameters; it is established on the basis of results. In fact, direct numerical methods could not be applied. However, the proposed method belongs to relaxation methods, which are preferred to solve these problems. Our approach consists of searching to converge toward minimal error in an iterative way, relying on an optimization model.

4.2 Resolution of the Optimization Problem

The problem of parameters' optimization is classified as an NP-difficult problem. The need to quickly find a good solution, promotes the appearance of rough or stochastic algorithms namely meta-heuristics [29].

In this context, meta-heuristics showed their performance for a wide variety of optimization problems. It gives more advantages compared to traditional algorithms, such as the ability to handle very high levels of complexity, the ability to accommodate several sets of issues and their application in many areas of the real world, starting from operational research, through engineering and artificial intelligence [30], where there is a need to optimize digital functions, and systems containing a large number of parameters to manage simultaneously.

Meta-heuristics use strategic research to explore more efficiently the search space, and focus on more promising areas. These methods start with an initial set of solutions or a starting population, after that, they examine, step by step, a sequence of solutions to reach (or approach) the problem optimal solution. Meta-heuristics have several advantages over traditional algorithms, the two most important advantages are simplicity and flexibility. It is often simple to implement them. Several examples are cited by Gandomi in [31]. These algorithms are very flexible and have the ability to treat problems with objective functions of various properties, whether continuous, discrete or mixed.

Following, the procedure employed to solve the problem of parameters' adjustment, applying ACO algorithm, is described. The ant colony optimization algorithm is a probabilistic technique, which can be similar to finding good paths through graphs.

4.3 The proposed ACO model

Ant colony optimization (ACO) is a technique used for various applications. ACO is inspired from the natural behaviour of real ants [32], it optimizes problems through guided search space [33]. An ant travels through paths and drops pheromone, and then its influence paths selection by other ants. In fact, this deposit works as a positive feedback by reinforcing good choices.

This characteristic of real ants' behaviour is used to solve difficult combinatorial optimization problems; therefore, ants build solutions based on artificial pheromone trails in a probabilistic way. The pheromone represents the main element of the ACO algorithm, while the transition and updating rules are based on its quantities.

The basic idea is to consider each domain as a food area. Each ant starts from D_1 and tries to reach D_M through the other areas and picks up food.

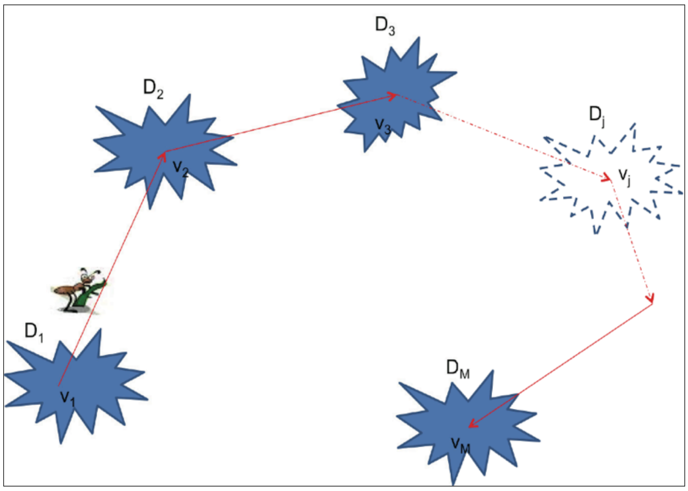


Figure 4. An ant path over domains Di

An ant is considered as a potential solution by building incrementally a path (the parameter values represent the gathered food by each ant). Values are selected, under a sequence, from predefined domains D_i (Figure 4). Then, the applied operators result, using the selected path, is evaluated by the objective function.

By this evaluation, each ant determines pheromone quantity to deposit on the path. Two problems are faced when selecting parameters' values:

- The choice of the domain to move to.
- The choice of a value within a domain.

Fixing a domain order corresponding to the operators' order can solve the first problem. For the second one, the choice is based on a probability calculation. Parameter values are taken as nodes (Figure 5). An ant constructs its path by choosing nodes in the proposed domains (one node in each domain). If the actual node is v_i, the choice of the next node v_j is based on a probability calculation (indexes i, j, k ... represent nodes v_i, v_j, v_k,...):

$$pr_{ij} = \begin{cases} \frac{(\tau_{ij})^{\alpha}\,(\eta_{ij})^{\beta}}{\sum_{k\in J(i)}(\tau_{ik})^{\alpha}\,(\eta_{ij})^{\beta}}, & if\ j\ \in J(i) \\ 0, & otherwise \end{cases} \quad (7)$$

Where J(i) represent nodes of the parameter P_j in D_j not yet chosen by an ant, which limits possible nodes for parameter P_j, η_{ij} is the heuristic information reflecting the desirability of choosing the value v_j in J(i) (discussed below) τ_{ij} is the current pheromone quantity between v_i and v_j. α and β are adjustable values.

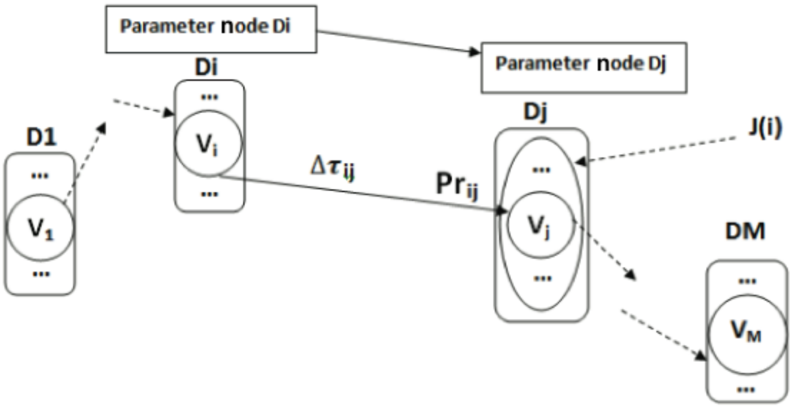


Figure 5. Probability that ant moves from value v_i (of node D_i) to a value v_j (of node D_j).

When an ant round of its travel over the nodes, a path is built. It is a sequence of parameter values corresponding to the considered operator's combination. Following this, operators are applied to the input image and deliver a result. To evaluate this path a measure is provided by the objective function standing for the error: Err (v_1, v_2, ..., v_M). Based on this measure the quantity of pheromone $\Delta\tau_{ij}$ is calculated, it is lodged on each link between two consecutive nodes v_i and v_j in the found path, this pheromone quantity will influence the construction of the next path. Pheromone quantity is calculated for each ant k according to:

$$\Delta\tau_{ij}^{k} = \frac{Q}{Err(v_1,v_2,\ldots,v_M)} \quad (8)$$

Q is an adjustable parameter, which represent pheromone intensity.

Pheromone evaporation, in each iteration, is recorded by:

$$\tau_{ij}(t+1) = (1-\ \rho)\tau_{ij}(t) + \sum_{k=1}^{s}\Delta\tau_{ij}^{k}(t) \quad (9)$$

Where **s** is the number of ants.

This process is iterative and parallel, where in each iteration, all ants construct paths and must stop if there is a convergence, or after a fixed number of iterations. Below we describe the algorithm used in this approach for each combination of operators:

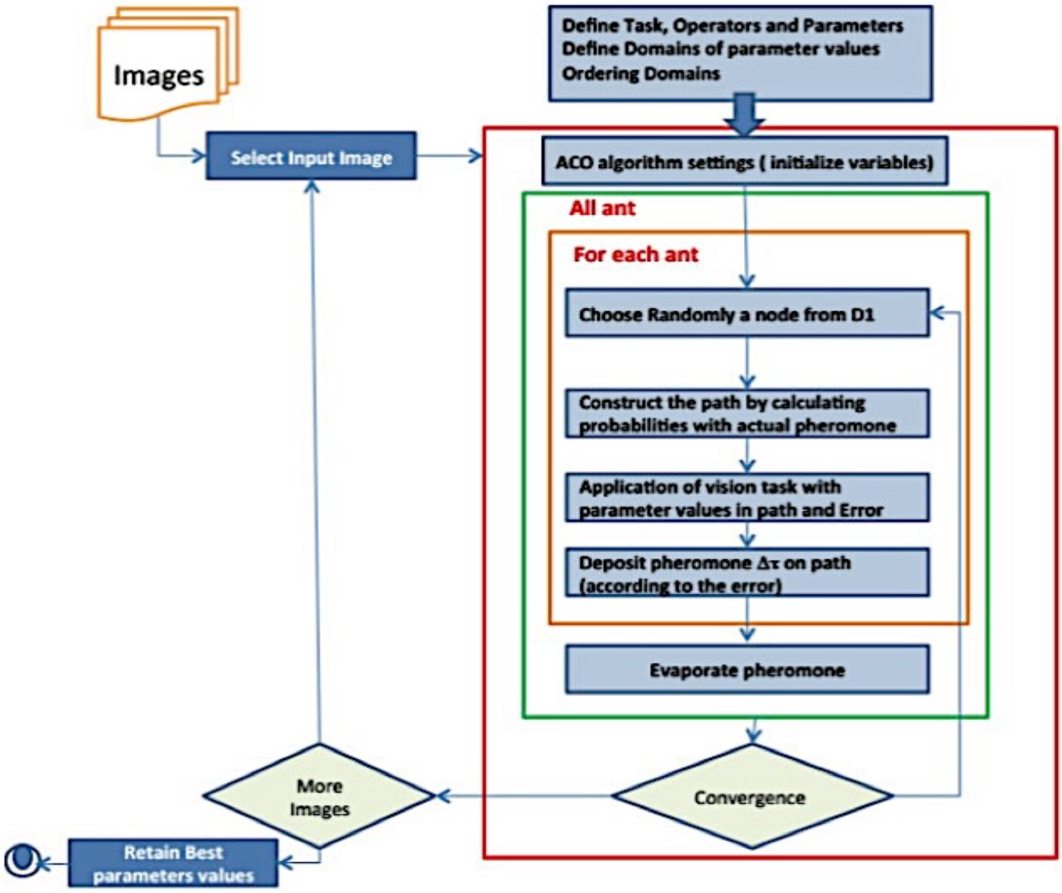


Figure 6. ACO algorithme used in this approach for each operators' combination

5 Experiments

In the previous section the approach is described, we are solving the problem of choosing optimal operators and the optimal values of their parameters in a vision task. The approach we presented theoretically is applicable to any vision task that needs operators' selection or parameters adjustment or both of them. Along this section, we have taken over some applications already treated and our approach is tested on several tasks of image processing, first one is about text recognition and aspect inspection related to tickets label on industrial products. The second one is about mechanical objects verification and last one is an investigation of surface texture.

In all these applications, we propose a vision system pre-work (preparation) which consists of choosing operators and adjusting their parameters in order to optimize results. In this pre-work, the user therefore proposes a set of possible operators for each operation, a range of parameters' values and a desired result (reference) for comparison. The desired result is given by the user with a margin of error and tolerance which can be images, calculations, etc.

In case the parameter setting does not offer the desired results, it is up to the user to change the possible operators and ranges of their parameters.

Our main contribution is to show that based on user data, the ACO approach offers the best solution for parameters setting by comparing obtained results to desired ones.

5.1 Case study 1: Text recognition

Let's consider a line producing bottles, we will apply the approach proposed above to find best operators and their best parameters' combination for a vision system controlling serial number on the bottles' sticker, this production line is a laboratory line designed for educational needs. In this experience a sticker is considered

improper if the serial number does not comply with the references, the serial number is different from a sticker to another, a list of references is provided by the expert in charge of the production line.
To validate our approach, we will carry out the same experience on images of one hundred stickers. These images are captured under different conditions (lighting, noise…), the percentage of good and improper stickers detected would give us an overview on the system recognition rate. So the system performance would be calculated based on this measure. To widen this quality system, we can add supplementary phases such as colours control, sticker position control, control of liquid level … this is possible using the approach proposed in this article, feasible operators and their possible parameters for each phase must be specified.

(a) (b)

Figure 7. (a) Production line of bottles using a vision system for quality control. (b) The sticker image.

5.1.1 Fixing tasks, operators and parameters:

The process of aspect inspection consists of detecting serial number, and making sure that it contains the correct one. Figure 8 shows different phases performed to achieve this vision task.

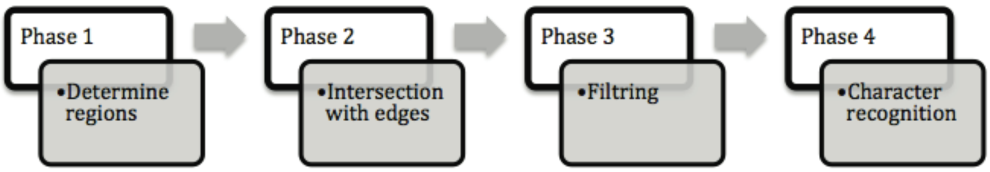


Figure 8. Different phases of the vision task.

Figure 9 illustrates operators used in each phase:

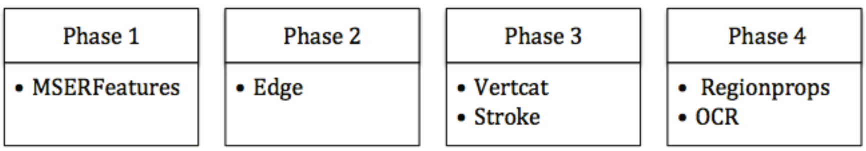


Figure 9. Operators' candidates for each phase.

This vision task will be implemented over four phases. Each one is concerned with a set of operators: MSERFeatures used in first phase, allows finding stable regions, phase two give out finding edges, which identifies points where the brightness has discontinuities, and then intersect it to Mserregions. In phase three we use two operators Vertcat and Strocke, the first one allows to remove the remaining connected components using their region properties, the second one is a discriminator for the text in images, it is useful to remove regions where the stroke width has exhibited too much variation since characters in most languages have a similar stroke width. Phase four, allows finding text regions and recognize text letters using Regionprops to determine text regions and OCR to recognize text. Each one of these operators, except OCR, has some parameters to be fixed depending on the image we are working on. Figure 10 summarizes the operators' parameters used to accomplish this vision task.

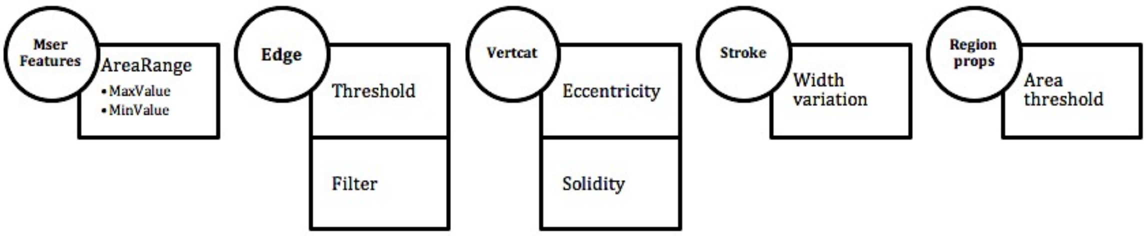


Figure 10. Adjustible parameters for each operator.

To find out the parameter values optimal combination, a range of values for each one of parameters is given up; a list of possible values of each parameter is provided in table 1. The list of operators and all possible values of their parameters, with image in figure 6(b) and reference serial number, constitute the input data provided to the ACO algorithm.

Table 1: Values candidates for each parameter

Operators	MserFeatures		Edge		Vertcat		Strock	RegionProps
	Max	Min	Threshold	Filter	Eccentricity	Solidity	Width variation	Area threshold
Values candidates for each parameter	10	2000	0,001	Canny	0,970	0,1	0,3	10
	20	3000	0,002	Log	0,975	0,2	0,4	20
	30	4000	0,003	Prewitt	0,980	0,3	0,5	30
	40	5000	0,004	Sobel	0,985	0,4	0,6	40
	50	6000	0,005		0,990	0,5	0,7	50
	60	7000	0,006		0,995	0,6	0,8	60
						0,7	0,9	70

5.1.2 Objective function

The ground truth defines tightly bound text boxes; a good detection algorithm would ideally produce similarly tight boxes. To evaluate the accuracy and tightness of our approach output, the pixel areas of the text region in the ground truth are matched with the detected text regions. The evaluation is thus a pixel-by-pixel comparison of our approach output with the ground truth.
During evaluation, each pixel in the test images is placed into one of three categories:

- Detection: Pixels belonging to text regions in the ground truth and regions identified as text by the algorithm.
- False Alarm: Pixels identified by the detection algorithm but not belonging to text regions in the ground truth.
- Missed Detection: Pixels belonging to the text regions in the ground truth and not identified by the algorithm.

The performance of our approach is quantified by its recall and precision, where:

$$\text{Recall} = \frac{detects}{detects+missed\ detects} \quad (10)$$

$$\text{Precision} = \frac{\text{detects}}{\text{detects + false alarms}} \quad (11)$$

The error considered and minimised through this experience is:

Error = (1-Recall) + (1-Precision) (12)

Note that this pixel-level evaluation is very strict. Most actual applications would not require such precise localization. However, our pixel-level criteria provide an easily measurable basis by which the relative performances of algorithms may be compared.

5.1.3 Optimal parameters

Fixing parameters for an optimization algorithm is a very important step before proceeding to parameter optimization of a vision task. We rely on literature to fix these parameters in this part. The ACO algorithm is applied using 30 ants, and based on research studies of ACO parameter selection on particular problems [34] [35], pheromone intensity Q is fixed to 100, evaporation coefficient ρ to 0,5, α to 1, β to 5 and η to 1. We consider that the desirability to choose the next node (i.e. a parameter value) is equivalent for all nodes; a more detailed study based on statistics may improve convergence time. We run the algorithm for 20 iterations, for the only operator's combination of this task: [MserFeatures, Edge, Vartcat, Strock, Regionprops, OCR, strcmp]. The optimal parameter values found are summarized in table 2, and Figure 11 relieve the error evolution across iterations. We notice that, at first error was considerable but over time ants choose values more marked by pheromone, so error decreases over iterations and becomes almost stable in minimal value.

Table 2: Optimal parameters values

Parameter	Optimal Value
Area-Range	[10,6000]
Filter	Canny
Threshold	0,04
Eccentricity	0,995
Solidity	0,2
Width-variation	0,9
Area-threshold	10

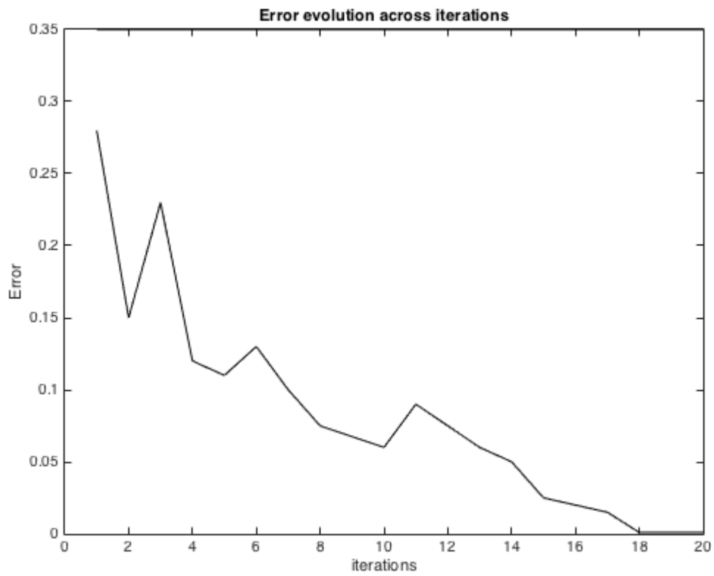


Figure. 11. Error evolution across iterations.

Images below (Figure 12) demonstrate experimental results of each operator, using optimal parameters. We detect three different regions. The first one contains the logo, on which an other procedure is necessary to control its similarity to the reference, we will focus on the second and third region containing a text, which we recognize using OCR and compare to references. The final error rate represents the sum of the two error rates calculated over the two text regions found (described in region 5.1.2).

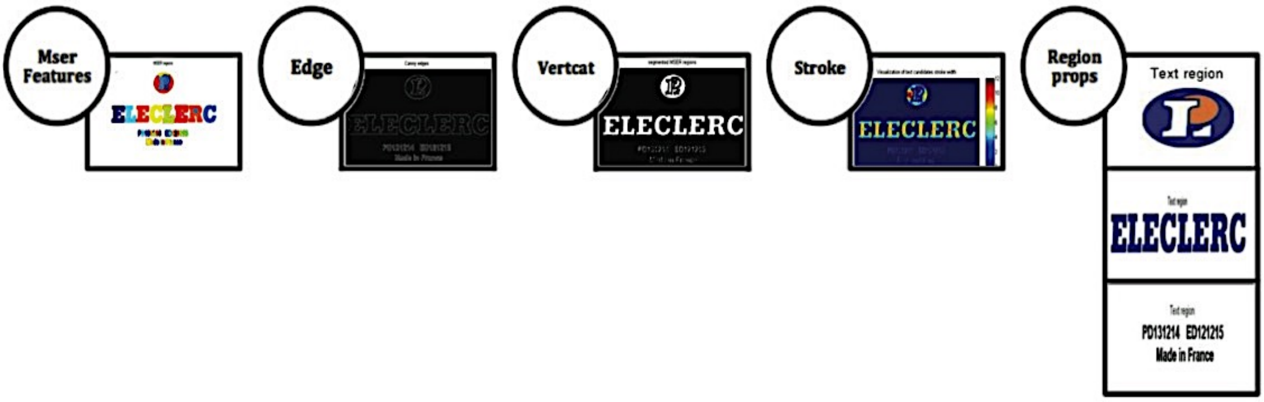


Figure. 12. Results of each operator application

The last step, after finding the best parameters' combination, is to compare text extracted to the reference using the operator "Strcmp". This operator returns a result of logical data type; it is true if the two texts compared are identical. This last part allows quantifying the vision system recognition rate.
Using the combination of best operators and their best values of parameters, the same experience was done on 100 bottles' images; text extracted from each sticker was compared to its reference (i.e. its unique serial number). No tolerance is accepted for this experience, no differences were noted for results extracted and references. To further more validate this 100% efficiency of our proposed vision system, more experiments most be done on real production line data. But these experiences are good enough to say that the optimisation method proposed in this work is very efficient for proposing best combination of vision operators and their best parameter values to be used in a vision system.

5.2 Case study 2: Mechanical objects

Let's consider a production line of pinions; a configuration of a vision system will be done in order to control the circle's diameter. A comparison between measurements given by the vision system and references would be done. It is a laboratory data; a tolerance rate of ±0,01 would be accepted (data given by the expert system).

The process of measurement's extraction consists of detecting circular objects and extracting diameters. This task is made of three phases, each one of these phases is concerned with a set of operators: phase one consists of filtering noises of the image, two operators can be used here Medfilt2 and Imgaussfilt, phase two try detecting interior circles and extract their diameters, finally phase three detect exterior circles and extract their diameters, the same operator Imfindcircles is used in phases two and three, but with different parameters. Figure 13 summarizes operators used to accomplish this vision task:

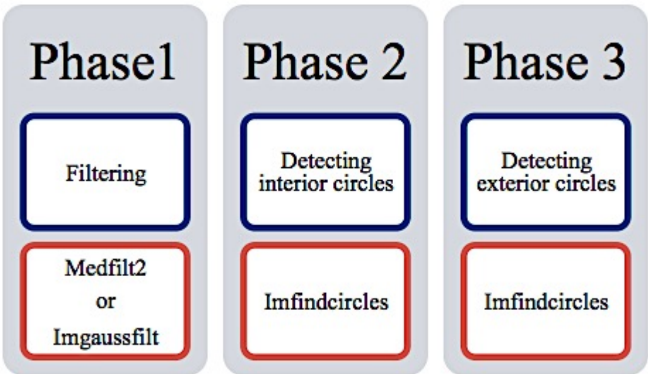


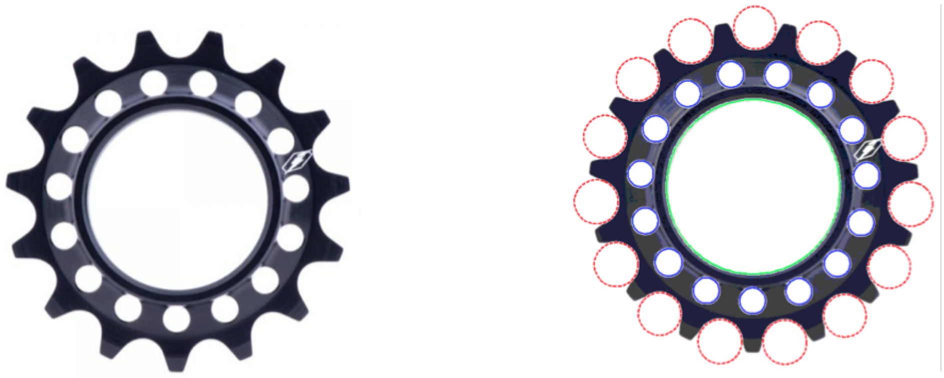
Figure. 13. Operator's candidates of each phase.

Each one of these operators has some parameters to be fixed depending on the image we are working on. To find out the parameter values optimal combination, a range of values for each parameter is given up. Table. 3 provides a list of operator's parameters and their possible values:

Table 3. Operators' candidates for each phase and their parameters with possible values for each one

	Phase 1		Phase 2					Phase 3				
Operators	Medfilt2	Imgaussfilt	Imfindcircles					Imfindcircles				
Parameters	Filtersize	Standard deviation	Radius range (Min < Max)		Object polarity	Sensitivity	Edge threshold	Radius range (Min < Max)		Object polarity	Sensitivity	Edge threshold
Possible values	1	0,1	Min	Max	Bright	0,3	0,2	Min	Max	Bright	0,3	0,2
	3	0,3	20	25	Dark	0,4	0,3	28	40	Dark	0,4	0,3
	5	0,5	25	30		0,5	0,4	38	45		0,5	0,4
	7	0,7	30	35		0,6	0,5	45	50		0,6	0,5
		0,9	35	40		0,7	0,7				0,7	0,7
			40	45		0,8	0,8				0,8	0,8
			45	50		0,9	0,9				0,9	0,9

Figure 14 (a) shows pinion concerned and Figure 14 (b) shows interior and exterior circles found.



(a) (b)

Figure. 14. (a) Pinion concerned. (b) Circles found using ACO approach.

Table 4 retrace extracted measurements compared to references, all measurements extracted are equal to references except one circle. The error is equal to -0,03, which is a tolerable value.

Table 4. Measurements extracted from figure 14, compared to their references

diameters extracted for interior circles (in Blue)	References	Error	diameters extracted for exterior circles (In Red)	References	Error
30	30	0.0	45	45	0.0
. . .	. . .	. . .	. . .	. . .	. . .
29,997	30	0.003	45	45	0.0

Table. 5 resume best operators’ combination and best parameters' values:

Table 5. Best operators' combination and best parameters' values.

	Phase 1	Phase 2					Phase 3				
Best Operators	Imgaussfilt	Imfindcircles					Imfindcircles				
Best Parameters	Standard deviation	Radius range (Min < Max)		Object polarity	Sensitivity	Edge threshold	Radius range (Min < Max)		Object polarity	Sensitivity	Edge threshold
Best values	0,3	Min	Max	Bright	0,9	0,7	Min	Max	Bright	0,9	0,7
		20	25				28	40			

The same experience was conducted on 100 images of pinions, using the combination of best operators and their best parameters’ values found above, a tolerance of 0,01 was acceptable. On the 100 images, just one piece was deficient. The difference between diameter extracted by our system and the reference exceeds the tolerance rate considered in this experience. By measuring diameters in real pinion in matter, it was really deficient and the measure extracted by our system is correct. This experience proves the efficiency of ACO method to optimize operators and parameters for image processing algorithms used for quality control.

5.3 Case study 3: Investigation of surface texture

This third and last case study will be about inspection of wood surface (refer to [36] for data provenance or more details), since the quality of surface influences the components suitability for a specific application, more attention had been given by researchers to measure the surface quality accurately. A reference image is given to compare segmentation quality, after finding the best segmented image and so the optimal operators’ and their parameters combination, a measure is taken manually using an interactive Matlab tool.

This task would be made of three phases:

First: Convert the RGB image into greyscale image with the range of 0 to 255 scales with the help of the operator rgb2gray.

Second: Enhancement of greyscale image by using a histogram sliding which adds constant brightness to all pixels within the image, using the operator imadjust. The imadjust has three parameters to be adjusted:

– Contrast limits of input image, specified as a two-element numeric vector (minimal value and maximal value).

– Contrast limits of the output image, specified as a two-element numeric vector (minimal value and maximal value).

– Gamma, the shape of curve describing relationship of input and output values.

Third: Use of High pass filter to enhance the edges of the Image, by eliminating the low frequency components using high pass filter. The operator imsharpen is used in this phase, with three parameters to be adjusted:

– Radius: a standard deviation of the Gaussian low pass filter, which controls the size of the region around the edge pixels that is affected by sharpening.

– Amount: The strength of the sharpening effect.

– Threshold: Minimum contrast required for a pixel to be considered an edge pixel, specified as a scalar in the range [0 1]. Higher values (closer to 1) allow sharpening only in high-contrast regions, such as strong edges, while leaving low-contrast regions unaffected. Lower values (closer to 0) additionally allow sharpening in relatively smoother regions of the image. This parameter is useful in avoiding sharpening noise in the output image.

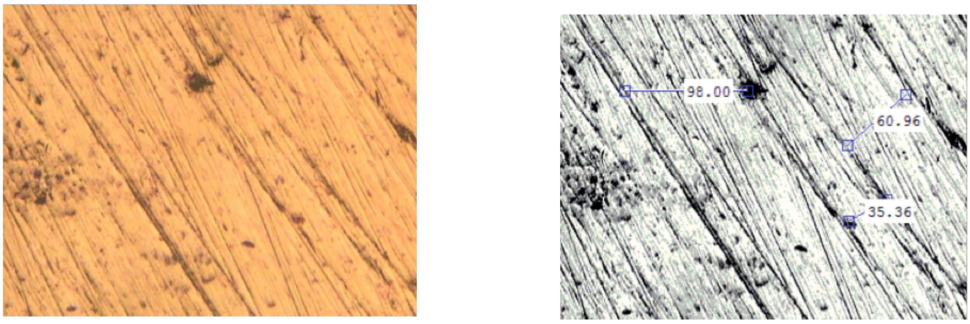
Calculating the pixel distance between two adjacent deep scratches, is done with the help of the operator imdistline, which creates a distance tool with endpoints located at the locations specified by two vectors x and y. At last these measurements are compared to references to further validate our approach.

Table. 6 resume operators and their parameters used in this task, a range of possible values would be given for each parameter.

Table 6. Operators candidates for each phase and their parameters with possible values for each parameter.

	Phase 1	Phase 2					Phase 3		
Operators	Rgb2gray	Imadjust					Imsharpen		
Parameters	No parameters to be adjusted	Contrast limit input (Min < Max)		Contrast limit onput (Min < Max)		Gamma	Radius	Amount	Threshold
		Min	Max	Min	Max				
Possible Values	**	0,1	0,1	0,1	0,1	0,2	0,1	0,5	0,2
		0,2	0,2	0,2	0,2	0,4	0,5	1	0,3
		0,3	0,3	0,3	0,3	0,6	1	1,5	0,5
		0,4	0,4	0,4	0,4	0,8	1,5	2	0,6
		0,5	0,5	0,5	0,5	1	2		0,7
		0,6	0,6	0,6	0,6	1,2	2,5		0,8
		0,7	0,7	0,7	0,7	1,4			
		0,8	0,8	0,8	0,8	1,6			
		0,9	0,9	0,9	0,9	1,8			

Figure 15 (a) shows surface image and Figure 15 (b) shows measurement of pixel count between two deep scratches; measurements extracted from the image are similar to references.



(a) (b)

Figure 15. (a) Surface image. (b) Measurement of pixel count between two deep scratches.

Table 7 resume best operators and their best parameters values:

Table 7: best operators' combination and best parameters' values.

	Phase 1	Phase 2					Phase 3		
Best Operators	Rgb2gray	Imadjust					Imsharpen		
Best Parameters	No parameters to be adjusted	Contrast limit input (Min < Max)		Contrast limit input (Min < Max)		Gamma	Radius	Amount	Threshold
Best Values	**	Min	Max	Min	Max	0,4	1	1,5	0,5
		0,3	0,7	0,4	0,6				

5.4 Discussion and ACO algorithm settings

This part represents a discussion about ACO algorithm parameters; we will carry on some experiences to adjust these parameters in order to see their influence on the result quality. The parameter η (fixed at 1 in our experiments) can be used to express dependence, constraints or hope between nodes. Many heuristic can be developed in this field, such as limiting a set of nodes regarding the fixed previous node or using a weighted graph on nodes, etc. We have studied the number of ants, pheromone intensity and the evaporation coefficient influence on the error value by varying them separately. The evolution of each parameter is shown in Fig. 16, 17 and 18.

5.4.1. Number of ants

Selecting the proper number of ants is a critical step because it affects the algorithm performance. The effect of the number of ants is shown in Fig. 16.

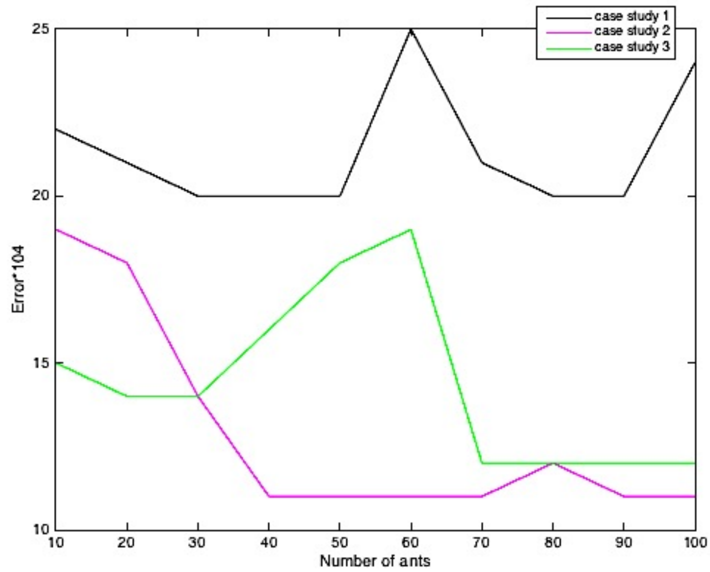


Figure 16. Calculated Error of ACO with increased number of ants.

5.4.2. Pheromone intensity

The role of pheromone intensity Q is to obtain the global solution to the problem with an appropriate evolution speed. Fig. 17 shows the error performance for evaporation intensity different values.

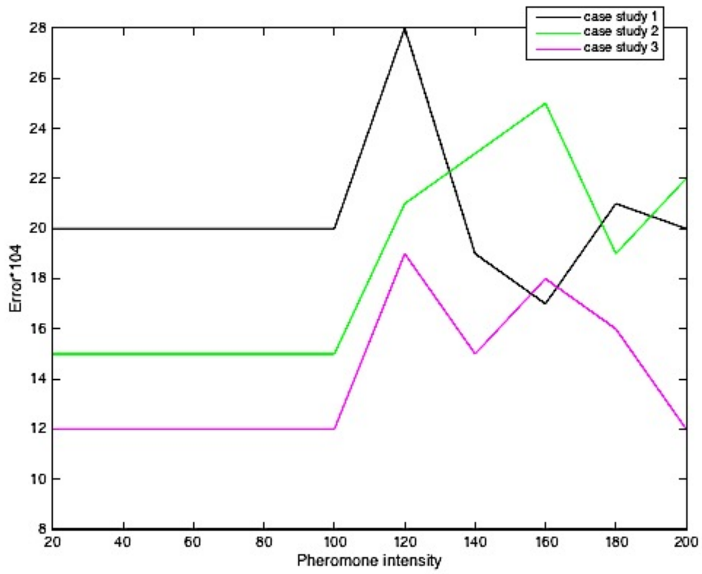


Figure 17. Calculated Error of ACO with increased Pheromone intensity values.

5.4.3. Evaporation coefficient

Evaporation coefficient allows decreasing uniformly all pheromone values. It is a form of forgetting which avoid a too rapid convergence. Fig. 18 shows the ACO model performance with different evaporation coefficient values.

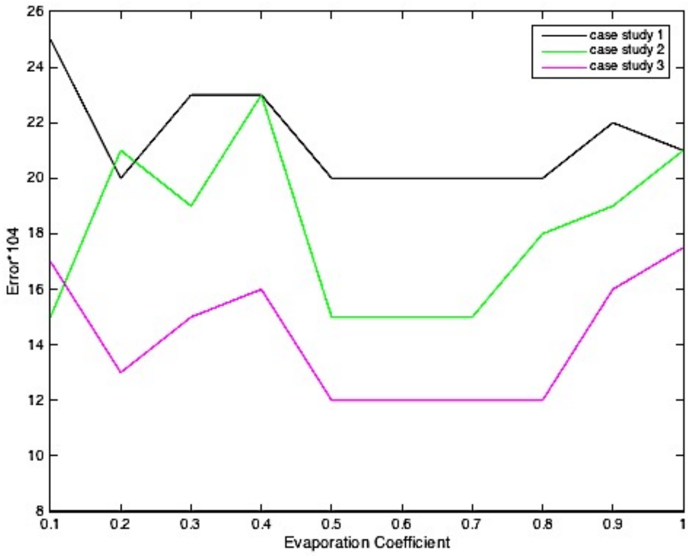


Figure 18. Calculated Error of ACO with increased Evaporation coefficient values.

It may be noted here that: the evolution of the error is not stationary according to the three figures above. We can simply locate the minimal error interval and choose the corresponding values. Thus, table 8 shows the ACO algorithm parameters' values for the three cases of study, based on the analysis above and previous work in literature [35] [37]. Actually, there is a slight error decrease regarding the values used in Section 5.3, Section 5.4 and Section 5.5 (first trial) and a slight modification of the optimal values of operators' parameters (Table. 9), but not too significant.

Table 8. Parameters setting of ACO model for each one of the three cases of study

Parameter	Values for case 1	Values for case 2	Values for case 3
Number of ants	30	40	70
Pheromone intensity Q	100	100	100
Evaporation coefficient ρ	0,5	0,5	0,5

Table 9. Optimal parameters values

Parameters case1	Optimal Values	Parameters case2	Optimal Values	Parameters case3	Optimal Values
Area-Range	[10,6000]	Standard deviation	0,3	Contrast Limit input	[0.2,0.7]
Filter	Canny	Radius range	[20, 25]	Contrast Limit input	[0.4,0.8]
Threshold	0,06	Object polarity	Bright	Gamma	0.6
Eccentricity	0,995	Sensitivity	0,9	Radius	1,2
Solidity	0,1	Edge threshold	0,7	Amount	1,5
Width-variation	0,9	Radius range	[28 40]	Edge threshold	0,7
Area-threshold	10	Object polarity	Bright		
		Sensitivity	0,92		
		Edge threshold	0,65		

To go beyond this study, we use a data set of 100 images and try to detect and recognize text on these images, using the same experimental setup described above. Figure 19 shows the error rates obtained by the proposed ACO approach with contrast to some other techniques presented in [3, 4, 10, 9].

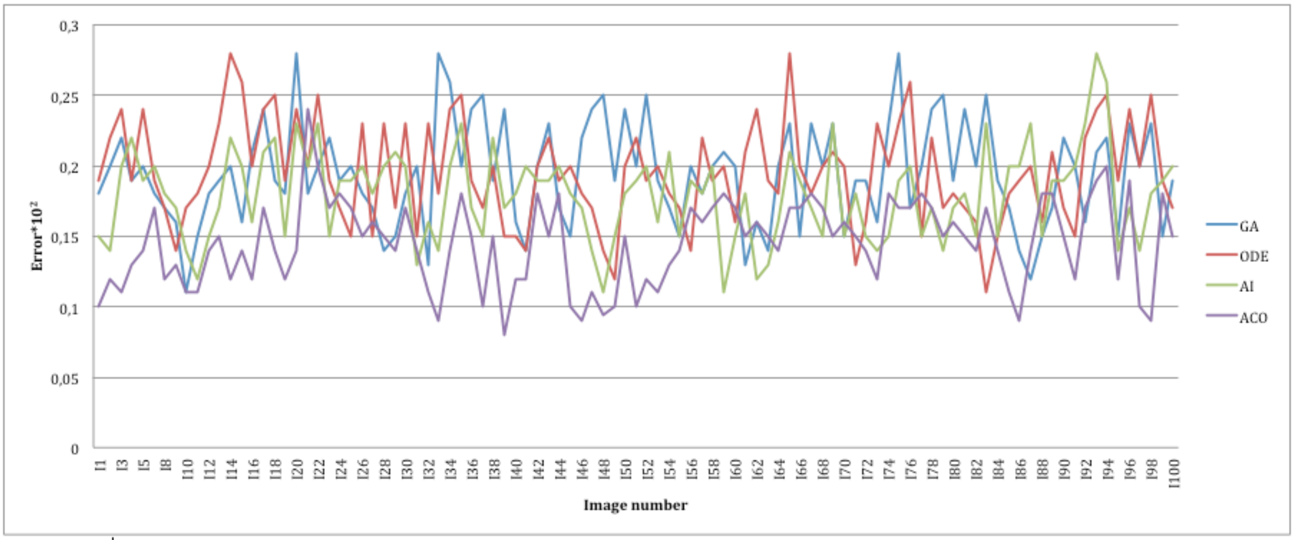


Figure 19. Plot of error rate in percentage with different methods. GA is genetic algorithm, ODE denotes orthogonal design of experiments, AI means artificial intelligence and ACO denotes the proposed ACO approach in this article.

The proposed method achieves almost the best results corresponding to minimal error rates on the sample of 100 images. Actually, when using genetic algorithm to detect the best operators and their best parameters' values we have an average of error rate of 0,1924 on the 100 images, it is a little more important using the orthogonal design of experiments since it is equal to 0,1935. On another side artificial intelligence gives promising results considering that the average of error rate on 100 images is 0,1776, and when using the proposed ACO approach, the error rate average is equal to 0,1434 representing a very promising result.

The examples treated here validate practically our approach for images in quality control domain. This approach constitutes a new general and express way of reasoning for any vision task that requires the right choice of operators and the right adjustment of parameters, since it is based on easy mathematical operations via pheromone model.

6 Conclusion

The choice of appropriate operators to apply and the values of their parameters for performing a vision task is a very big challenge for users. In this work, an automated method is presented to optimize the parameter values of image processing algorithms in quality control. Our approach decides automatically which operator is most appropriate to use and adjust its parameters based on ACO algorithm.

This work demonstrates this approach performance, and concludes that the use of ACO algorithm for automating operators' choice in image processing and adjusting parameters is very promising. It must be remembered that the wish to choose nodes was fixed to one, so selecting the next parameter value in an ant travel is uniform. Although it represents a parameter, which must be studied more and it may improve the results.

References

[1] C.Onwubiko, Introduction to Engineering Design Optimization. Prentice Hall, 2000.

[2] L.Benchikhi, M.Sadgal, A.Elfazziki, and F.Mansouri, “A Discrete Particle Swarm Optimization to Estimate Parameters in Vision Tasks” International Journal of Advanced Computer Science and Applications(IJACSA), 7(1), 2016. http://dx.doi.org/10.14569/IJACSA.2016.070128
[3] I.Qaffou, M.Sadgal, and A.Elfazziki, A multi-agents architecture to learn vision operators and their parameters. International Journal of Computer Science. pp: 140, 2012.
[4] S.Treuillet, D.Driouchi , and P.Ribereau, Adjustment of parameters in an image processing chain by an experimental 2^{k-p} factorial designs. Traitement du signal, pp: 141–155, 2004.
[5] A.J.Pretorius, D.Magee, D.Treanor, and R.A.Ruddle, Visual parameter optimization for biomedical image analysis: A case study. In: Proc. SIGRAD, November, 2012.
[6] A.Ansari and F.Hou, Computational Intelligence for Optimization. Springer, 1997.
[7] B.Yukun, H.Zhongyi, and T.Xiong, A PSO and pattern search based memetic algorithm for svms parameters optimization. Neurocomputinh, pp: 98–106, 2013. DOI: https://doi.org/10.1016/j.neucom.2013.01.027
[8] T.Sag and M.Cunkas, Color image segmentation based on multiobjective artificial bee colony optimization. Applied soft computing, pp: 389–401, 2015. DOI: https://doi.org/10.1016/j.asoc.2015.05.016
[9] P.Kumsawat, K.Attakitmongcol, and A.Srikaew, A new approach for optimization in image watermarking by using genetic algorithm. EEE Trans. Signal Process, pp: 4707–4719, 2005. DOI: 10.1109/TSP.2005.859323
[10] L.Benchikhi, M.Sadgal, and A.Elfazziki, An optimization approach of parameters in image processing based on PSO: case of quality control. In: Proc. The International Conference on Industrial Engineering and Systems management, Morocco, 2013.
[11] A.Colorni, M.Dorigo, and V.Maniezzo, Distributed optimization by ant colonies. In: Proc. 1st European Conference on Artificial Life, pp: 134–142., 1991.
[12] M.Dorigo, Optimization Learning and Natural Algorithms. PhD thesis, Politecnicodi Milano, Italy 1992.
[13] J.E.Bell and P.R.McMullen, Ant colony optimization techniques for the vehicle routing problem. Advanced Engineering Informatics, pp : 41–48, 2004. DOI: https://doi.org/10.1016/j.aei.2004.07.001
[14] P.Lin, J.Zhang, and M.A.Contreras, Automatically configuring ACO using multilevel paramils to solve transportation planning problems with underlying weighted networks. Swarm and Evolutionary Computation, pp: 48–57, 2014. DOI : https://doi.org/10.1016/j.swevo.2014.10.006
[15] C.Blu, Hybridizing ant colony optimization with beam search: an application to open shop scheduling. Computers and Operations Research, pp : 1565–1591, 2005. DOI : https://doi.org/10.1016/j.cor.2003.11.018
[16] K.Socha and M.Dorigo. Ant colony optimization for continuous domains. European Journal Research, pp : 1155–1173, 2008. DOI : https://doi.org/10.1016/j.ejor.2006.06.046
[17] C.Mao, L.Xiao, X.Yu, and J.Chen, Adapting ant colony optimization to generate test data for software structural testing. Swarm and Evolutionary Computation, pp: 23–36, 2015. DOI : https://doi.org/10.1016/j.swevo.2014.10.003
[18] J.Tian, W.Yu, and S.Xie. An ant colony optimization algorithm for image edge detection. Evolutionary Computation, IEEE World Congress on Computational Intelligence, pp : 751–756, 2008. DOI : 10.1109/CEC.2008.4630880
[19] R.Laptik and D.Navakauskas. Application of ant colony optimization for image segmentation. Electronics and electrical engineering, 2007.
[20] M.E.Lee, S.Kim, W.Cho, and S.Park, Segmentation of brain MRI images using an ant colony optimization algorithm. Ninth IEEE International Conference on Bioinformatics and BioEngineering, 2009. DOI : 10.1109/BIBE.2009.58
[21] A.R.Malisia and H.R.Tizhoosh, Image thresholding using ant colony optimization. In: IEEE 2006 Canadian Conference on Computer and Robot Vision; 2006. DOI : 10.1109/CRV.2006.42
[22] J.Zhang, J.Zhou, H.He, and H.Li, Image edge detection using quantum ant colony optimization. International journal of digital content technology its applications, 2012.
[23] A.V.Baterina and C.Oppus, Image edge detection using ant colony optimization. Wseas transactions on signal processing, 2010.
[24] H.R.Kanan, K.Faez, and M.Hosseinzadeh, Face recognition system using ant colony optimization-based selected features. IEEE Symposium on computational Intelligence in Security and Defense Applications, pp: 57–62, 2007.
[25] A.Kaven, B.Hassani, S.Shojaee, and S.M.Tavakkoli, Structural topology optimization using ant colony methodology. Engineering Structures, pp: 2559–2565, 2008. DOI : https://doi.org/10.1016/j.engstruct.2008.02.012
[26] X.Zhang, X.Chen, and Z.He, An ACO-based algorithm for parameter optimization of support vector machines. Expert systems with applications, pp: 6618–6628, 2010. DOI : https://doi.org/10.1016/j.eswa.2010.03.067

[27] K.Loukhaoukha, J.Y.Chouinard, and M.H.Taieb. Optimal image watermarking algorithm based on lwt-svd via multi-objective ant colony optimization. Journal of Information Hiding and Multimedia Signal Processing, 2011.
[28] B.Chan, L.Chen, and Y.Chen. Efficient ant colony optimization for image feature selection. Signal processing, pp: 1566–1576, 2013. DOI : https://doi.org/10.1016/j.sigpro.2012.10.022
[29] C.Blum, A.Roli. Metaheuristics in combinatorial optimization, Overview and conceptual comparison. ACM Computing Surveys (CSUR), 35(3):268-308, 2003. DOI : 10.1145/937503.937505
[30] X.Yang, Z.Cui, R.Xiao, A.Gandomi, M.Karamanoglu. Swarm intelligence and bio-inspired computation: theory and applications. Newnes, 2013.
[31] A.Gandomi, X.Yang, A.Alavi. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Engineering with Computers, 29(1):17-35, 2011. DOI: 10.1007/s00366-011-0241-y
[32] M.Dorigo, V.Maniezzo, and A.Colorni. The ant system: Optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern, pp: 29–41, 1996. DOI : 10.1109/3477.484436
[33] M.Dorigo and T.Stutzle. Ant colony optimization. MIT Press, 2004.
[34] M.Samrout, R.Kouta, F.Yalaoui, E.Chatelet, and N.Chebbo, Parameter's setting of the ant colony algorithm applied in preventive maintenance optimization. Journal of Intelligent Manufacturing, pp: 663–677, 2007. DOI : 10.1007/s10845-007-0039-3
[35] H.B.Duan, D.B.Wang, and X.F.Yu, Research on the optimum configuration strategy for the adjustable parameters in ant colony algorithm. Journal of Communication and Computer, pp: 32–35, 2005.
[36] M.Anthony, A.Srivania, Investigation of surface texture using image processing. 12th Global Congress on Manufacturing and Management, 2014. DOI : https://doi.org/10.1016/j.proeng.2014.12.348
[37] F.Yalaoui, E.Chatelet, N.Chebbo, M.Samrout, R.Kouta, Parameteres setting of the ant colony algorithm applied in preventive maintenance optimization. Journal of Intelligent Manufacturing, pp: 663–677, 2007. DOI : 10.1007/s10845-007-0039-3