Initialization of Optimization Methods in Parameter Tuning for Computer Vision Algorithms

Andrea Bessi, Daniele Vigo, Vincenzo Boffa and Fabio Regoli

Abstract Computer Vision Algorithms (CVA) are widely used in several applications ranging from security to industrial processes monitoring. In recent years, an interesting emerging application of CVAs is related to the automatic defect detection in some production processes for which quality control is typically performed manually, thus increasing speed and reducing the risk for the operators. The main drawback of using CVAs is represented by their dependence on numerous parameters, making the tuning to obtain the best performance of the CVAs a difficult and extremely time-consuming activity. In addition, the performance evaluation of a specific parameter setting is obtained through the application of the CVA to a test set of images thus requiring a long computing time. Therefore, the problem falls into the category of expensive Black-Box functions optimization. We describe a simple approximate optimization approach to define the best parameter setting for a CVA used to determine defects in a real-life industrial process. The algorithm computationally proved to obtain good selections of parameters in relatively short computing times when compared to the manually determined parameter values.

A. Bessi · D. Vigo (✉)
DEI, University of Bologna, Bologna, Italy
e-mail: daniele.vigo@unibo.it

A. Bessi
e-mail: andrea.bessi3@unibo.it

V. Boffa · F. Regoli
Pirelli Tyre S.p.A, Milano, Italy
e-mail: vincenzo.boffa@pirelli.com

F. Regoli
e-mail: fabio.regoli@pirelli.com

A. Sforza and C. Sterle (eds.), *Optimization and Decision Science: Methodologies and Applications*, Springer Proceedings in Mathematics & Statistics 217,
DOI 10.1007/978-3-319-67308-0_20

1 Introduction

The calibration of computer vision algorithms (CVAs) is a time consuming and critical step in the effective use of CVAs in many applications, such as the automated defect detection of pieces produced by an industrial plant. In this case, the final quality control check at the end of the production chain consists in the optical scan of the produced pieces by a set of several CVAs. Each of them is designed to detect a specific type of defect and its behavior is controlled by a large set of parameters, which influence the CVA sensibility and accuracy and must be determined to maximize its detection efficacy on specific types of images. For a general overview of automated defect detection see [7] (see also [5] for an example in the textile industry).

More precisely, given a set of images, the efficacy of the error detection is measured as a function of the positive and negative false ratios produced by the CVA with a specific parameter set. As in many other applications parameter tuning of the CVAs is, therefore, a crucial component for the overall efficacy of the system. To the best of our knowledge, no optimization method has been developed so far for parameter tuning in defect detection.

In the context of CVAs, the computation of the efficacy requires the application of the CVA to a training set of test images. This is typically a very time-consuming operation requiring several seconds per image, hence minutes or even hours for a significant training set. Therefore, approaches based on black-box function optimization (see, e.g., [2, 3]) must be used in this case. To this end, we developed a simple Sequential Approximate Optimization (SAO) algorithm (see, e.g., [4]) to identify the optimal parameter values for a CVA used to detect a specific error on the images. During the optimization process, the solutions iteratively found by the algorithm are evaluated by executing the target CVA on the training set of images. The comparison between the CVA outputs obtained on the images and their real defectiveness state produce the true/false positive index ratio for the solutions tested. Our goal is the determination of the optimal input parameter combination for the CVA, leading to the best possible false positive and negative ratios for each particular type of defect.

In Sect. 2 we describe in detail the characteristic of the problem under study. In Sect. 3 the structure of the proposed algorithm is given and in Sect. 4 we present the results of an experimental validation of the algorithm on data coming from a specific real-world application.

2 Problem Definition

The calibration of the parameters of a CVA is an optimization problem which can be described as follows. The variables to be optimized are the input parameters of the CVA which are assumed here to be continuous and associated with a lower and an upper bound) for their variation. The performance of the CVA is measured in terms of two independent indicators, namely the number of false-negative and false-positive in the solution, to be defined later.

More precisely, we have a CVA whose behavior depends on a subset I of parameters whose value has to be determined. For each parameter $i \in I$ we are given a lower and upper bounds l_i and u_i, respectively. For each parameter $i \in I$ let x_i be the decision variable which represents its value. Given solution $\vec{x}$ we can evaluate its quality by measuring the performance of the CVA on a training set S of images. To this end, let $f_p(\vec{x})$ be the number of false-positives returned by the CVA when applied to the set S with parameters $\vec{x}$, defined as the number of non-defective images which are classified as defective by the CVA. Similarly, let $f_n(\vec{x})$ be the number of false-negatives returned by the CVA, defined as the number of defective images which are classified as non-defective by the CVA. Finally, let α_p and α_n be two nonnegative weights associated with the two performance measures. The CVA Parameter Tuning Problem (CVAPTP) can be formulated as follows

$$(CVAPTP) \quad z = \min F(\vec{x}) = \{\alpha_p f_p(\vec{x}) + \alpha_n f_n(\vec{x})\}, \quad s.t.\ l_i \leq x_i \leq u_i \quad \forall i \in I. \quad (1)$$

3 A Sequential Approximate Optimization Algorithm

The Black Box Optimization (BBO) nature of the problem requires the use of an inference intelligence able to predict the objective function value of an unsampled solution to guide the search process. In the literature, such representation of the BB function it is referred to as the Surrogate Model (SM, see [8]) for which a large number of types and formulations were proposed.

As frequently done in the recent literature the SM is used within a Sequential Approximate Optimization (SAO) algorithm that iteratively updates the SM by adding the solution points that are determined at each iteration. This sequential approach preserves a certain simplicity but provides some important advantages. First, the rebuilding of the SM in order to capture the incoming information improves its reliability at each iteration. Second, it guarantees to perform a global optimization over the entire solutions domain, by reducing the possibility to being trapped into local optima. The general scheme of the simple SAO algorithm we adopted is depicted in Fig. 1.

The algorithm starts with the identification of the initial sample of solution points, used to initialize the SM. Then, for each sample point, the corresponding value of the objective function $F(\vec{x})$ in (1) is computed by applying the parameter values associated with the point to the CVA over the entire images test set. The incumbent solution is defined as the best solution found so far, hence it initially corresponds to the best sample. The SM is built from the current set of points $(\vec{x}, F(\vec{x}))$ and used to determine the next candidate solution. After this, the SM is interrogated in order to search for the best candidate solution possibly improving the incumbent. This phase is generally called *adaptive sampling* criteria. The process is iterated until termination criteria based on solution quality and running time are met.

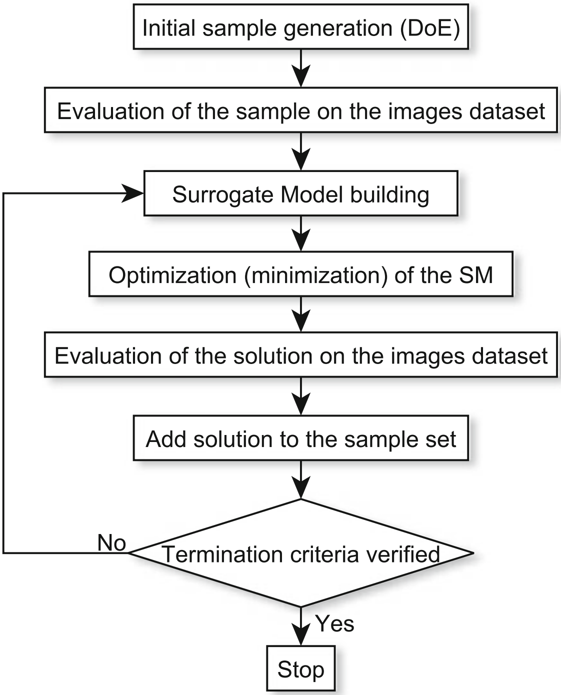
Initial sample generation (DoE)
Evaluation of the sample on the images dataset
Surrogate Model building
Optimization (minimization) of the SM
Evaluation of the solution on the images dataset
Add solution to the sample set
Termination criteria verified
No
Yes
Stop

Fig. 1 Outline of the sequential approximate optimization algorithm

As to the type of SM used, in this work due to the BBO nature of the problem we adopted a specific type of the so called Meshfree methods, named Radial Basis Function (RBF) interpolation techniques. These are relatively easy to construct and are widely used to approximate Black Box function responses. In RBF interpolation the model, $s(\vec{x})$, is defined as the sum of a given number K of radial functions ϕ:

$$s(\vec{x}) = \sum_{J=1}^{K} \gamma_j \phi(\|\vec{x} - \vec{x}_j\|) \tag{2}$$

where $\vec{x}_j, j = 1, \cdot, K$, is the set of sampled solution points representing the centers of the radial functions $\phi()$, γ is a vector of weights to be determined, and $\vec{x}$ is the unsampled point whose value has to be predicted. Regarding the radial functions ϕ, several type are available in literature, varying from parametrized to not parametrized ones. In our case, the best trade off between a simple construction of the SM and an acceptable reliability resulted with the use cubic basis function, that assume the form:

$$\phi(\|\vec{x} - \vec{x}_j\|) = \|\vec{x} - \vec{x}_j\|^3 \tag{3}$$

We refer the reader to [1] for an overview of Meshfree methods, and to [6] for an example of industrial use of cubic RBF.

As to the initial sample generation through which initialize the SM, several techniques exists in literature, typically referred to as Design of Experiment (DoE). Since in our problem the parameters x_j are not subject to other constraints besides the upper and lower bounds, classical DoE as the Factorial Design are suitable. In particular, a Full Factorial Design (FFD) permit to cover the entire domain space, selecting all the points of the grid generated by the discretization of each design variable (i.e., the parameters in our problem). This methodology is appropriate with the cubic RBF interpolation since it guarantees a sufficient reliability only inside the convex hull of the sampled points. However, in our case the computation of $F(\vec{x})$ is extremely time-consuming and using grids in which the parameter's values are discretized is not practically possible. For this reason, we decided to initialize our algorithm through a $|I|^2$ FFD, using just the domain vertices obtained with the lower and upper bounds of the parameters to be optimized. To improve the initial sample quality we also considered an initialization in which H additional random points selected inside the domain hypercube are considered. The adaptive sampling strategy that we adopt to perform the search of the candidate solution on the SM is the minimization of its predictor $s(\vec{x})$. To avoid to being trapped in a local minimum and perform an efficient search over all the domain, we implement a multi-start gradient descent algorithm and run it by using a discrete grid of starting points.

Finally, we terminate the algorithm after a maximum number of objective function evaluations or after a given number of non-improving iterations.

4 Experimental Validation

We applied our algorithm to the tuning of a CVA used to detect a specific type of defects on tyre images obtained in a real-world production environment. The training set is made up of 160 images for which the presence or absence of defects is known. Six parameters were selected as the target for the optimization. Each such parameter has a maximum and minimum value and a default value manually determined by the CVA designers. The behavior of the CVA with the default parameter values is used here as a benchmark reference to evaluate the performance of the optimized parameters set.

We tested the impact of three variants for the initialization step, leading to three different overall algorithms A_1, A_2, and A_3. In A_1 we used $H_1 = 100$ random points to initialize the algorithm. In A_2 the sample set is constituted by the 2^6 points of the simple FFD described in Sect. 3. Finally, in A_3, we added to the FFD set $H_3 = 36$ random internal points. The overall algorithm is run for a total of 200 objective function evaluations (including those for the initialization step). The gradient descent search for the candidate solution is performed from a 9^6 discrete grid fo points.

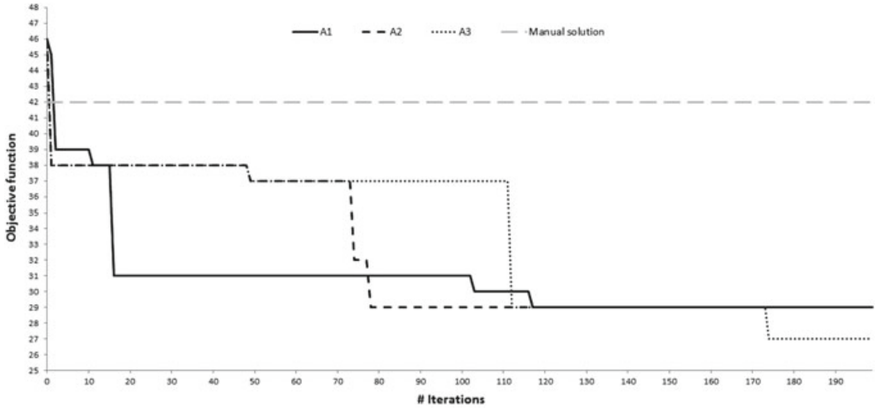

Fig. 2 Evolution of the proposed algorithms with weight combination (1,5) in comparison with the manual tuning

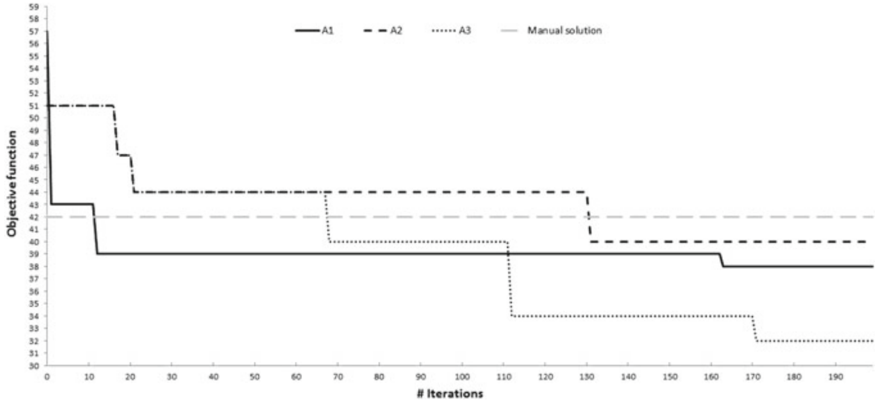

Fig. 3 Evolution of the proposed algorithms with weight combination (1,10) in comparison with the manual tuning

To account for possible different relative importance of false positives and false negatives in the defect detection, we considered two different pairs of weights in the objective function (1). Namely, we considered $(\alpha_p, \alpha_n) = \{(1, 5), (1, 10)\}$.

The results for the three algorithms are illustrated in Figs. 2 and 3 for the (1,5) and (1,10) weight combinations, respectively. The figures report the evolution of the objective function for each algorithm compared with the benchmark reference equal to 42 for both weight combinations. By observing the figures it clearly appears that all proposed algorithms generate better parameter combinations with respect to the manual ones. In particular, for the (1,5) case A_1,A_2 and A_3 produce solutions with value 29, 29 and 27, respectively, which are 31% and 36% better that then manual ones. For the (1,10) case, they find a solution with value 38, 40 and 32, which are

10%, 5%, and 24% better, respectively. In general, we can observe that the mixed initialization of A_3 provides better final results but the simple FFD of A_2 improves quite rapidly and may constitute a good alternative when less time is available.

References

1. Fasshauer, G.: Meshfree Approximation Methods with Matlab. World Scientific Publishing Co., Inc., River Edge, NJ, USA (2007)
2. Jones, D.: A taxonomy of global optimization methods based on response surfaces. J. Global Optim. **21**, 345–383 (2001)
3. Jones, D., Schonlau, M., Welch, W.: Efficient global optimization of expensive black-box functions. J. Global Optim. **13**, 455–492 (1998)
4. Kitayama, S., Arakawa, M., Yamazaki, K.: Sequential approximate optimization using radial basis function network for engineering optimization. Optim. Eng. **12**, 535–557 (2011)
5. Kumar, A.: Computer-vision-based fabric defect detection: a survey. IEEE Trans. Indust. Elect. **55**(1), 348–363 (2008)
6. McDonald, D., Grantham, W., Tabor, W., Murphy, M.: Global and local optimization using radial basis function response surface models. Appl. Mathemat. Model. **31**, 2095–2110 (2007)
7. Newman, T., Jain, A.: A survey of automated visual inspection. Comput. Vis. Image Understand. **61**(2), 231–262 (1995)
8. Queipo, N., Haftka, R., Shyy, W., Goel, T., Vaidyanathan, R., Tucker, P.: Surrogate-based analysis and optimization. Progr. Aerosp. Sci. **41**, 1–28 (2005)