Optimization of Configuration Parameter Set in Video Analysis

Jakub Król and Michał Grega
AGH University of Science and Technology, Kraków, Poland
Email: jakub.krol@poczta.fm, grega@kt.agh.edu.pl

***Abstract*—In our ongoing research we focus on detection of firearms and knives in video stream from Closed Circuit Video Television. During this research we came across a multiple-objective optimization problem. Our system operation depends on a set of launch parameters that differ between experiments. In this paper we describe and compare several automated optimization algorithms used for selection of these parameters. We apply those algorithms in order to select the best configuration parameters and compare them in the context of our system aiming at automated detection of dangerous tools.**

***Index Terms*—hyperparameter optimization, genetic algorithms, random search, Bayesian optimization, simulated annealing, firearms detection**

I. Introduction

In this paper we describe a solution, that was developed and tested while working on a system for automated detection of dangerous tools (such as firearms and knives) in the Closed Circuit Television (CCTV) streams1. Our system is designed to raise and alarm and alert the CCTV system operator if a person holding such an object is detected by out algorithms. Our requirements for the system is to (at highest priority) keep the number of false positives as low as possible (with target value at one for each 24 hours of a CCTV stream) and (at lower priority) maximize the number of dangerous event detections.

The system consists of several modules, of which two are crucial for this paper – the neural network module (denoted as H_NN) and the decision maker module (denoted as H_DM). The H_NN module is a deep neural network trained to analyze sub-images of a single frame from a CCTV stream. The sub-images are selected using a sliding window of a constant size. The H_NN returns, for each of the sub-images, the probability that the sub-image contains a dangerous object.

As it can be observed, in such approach the H_NN module utilizes only intra-frame information contained in each of the frames. The H_DM module, on the other hand, utilizes the inter-frame information and the fact, that the dangerous object is usually visible on multiple consecutive frames and should be located in generally similar location on the consecutive frames.

Manuscript received July 26, 2019; revised November 8, 2019.

One of the problems we had to tackle was how to select the optimal parameters that determine the operation of the H_DM module in order to meet the project requirements. Moreover, the parameter selection process had to be fast and highly automated, as in the research on the H_NN module we have experimented with numerous (over 40) different architectures, each requiring a separate set of H_DM parameters.

II. Problem Statement

Ideally the parameters chosen for a system should be optimal, however in some cases finding the optimum may be impossible or impossible in a reasonable time. Nonetheless automated optimization was required in order to find suboptimal parameter set over given training dataset.

For the H_NN module we have experimented with numerous architectures, each requiring its own suboptimal parameter set. The numerical results presented in this paper are based on one of those architectures – the MobileNet [1] enhanced by training on our own dataset (ex. [2]).

The H_NN returns a queue containing the coordinates of the sub-image with highest probability of containing firearms as well as that probability value. The H_DM module takes as an input that queue and returns another, this time only with coordinates of detection (or (-1, -1) value when no detection has been determined).

For the purpose of evaluation the results from the H_NN the H_DM requires 3 parameters:

- Maximum detections meaning how many subsequent frames in the video should contain positive detections. This parameter utilizes assumption, that a firearm or a knife should be visible on more than one subsequent frame.
- Maximum distance (measured in pixels) between sub-images on subsequent frames with those detections. This parameter utilizes assumption, that a firearm or knife should be detected in a given small radius between subsequent frames, as it should not move far at our recording speed of 25 Frames Per Second (FPS).
- Threshold meaning the minimum probability that given sub-image contains firearms, that defines the sensitivity of the H_DM module.

In order to meet the project requirements we proposed the weighted sum model [3], [4] for evaluating H_DM

doi: 10.18178/joig.7.4.130-133

module behavior. The selected weights reflect the importance of avoiding false alarms over the overall ability of the system to detect events and have been determined as a result of conducted tests. For the purpose of our project we define 'an event' as a series of continuous positive detections, after which there is a no detection period. Thus, we aim to maximize the function as defied in formula (1).

$$f = - 0.8 \cdot n + 0.2 \cdot g \tag{1}$$

where:

- f - H_DM's function value
- n - events detected on movies which don't contain firearms
- g - events detected on movies which do contain firearms

III. Related Work

The idea of choosing optimal hyperparameter set in video analysis has been refereed in articles such as [5] and [6] where authors used grid search with k-fold cross-validation in search for optimal SVM parameters. Also in [7] authors used the particle swarm optimization for that purpose. Lameski *et al.* [8] describe the role of grid search in SVM parameters tuning in preventing over-fitting.

This approach is also present in other domains. In [9] the authors give a review of decision making techniques of detecting heart diseases with parameters optimized by using a genetic algorithm. In [10] water demand forecasting system is optimized by using parallel global optimization method. In [11] the authors use Bayesian optimization for drug-target interaction prediction

The comparison of hyperparameter optimization is widely explored in such articles as [12]. In [13] and [14] a comparison of different evolutionary approaches to hyperparameter optimization is provided. The authors of [15] did a performance comparison for the purpose of utilization in wireless sensor networks. Moreover, Google provides its own internal service for blackbox optimization [16].

IV. Algorithms

The initial approach featured first manual search then grid search — the entire system was launched for every movie in the test set and for every parameter set in a grid in order to determine the best solution. Because of the size of training set and the processing time of one movie this solution was limited to very small grid of parameters determined mostly by intuition and even then the time of computing was measured in days.

Because of the modular construction of our system it was possible to run only selected modules easily. To make use of that feature we had run our system excluding the H_DM module and had logged the output from the H_NN to a file. The following approaches pick the parameters by running only the H_DM itself taking the input from that log file. The major time improvement was made the H_NN module is the slowest part of the system and it had been run only once for the training dataset.

The first of the optimization approaches was to use random search as it has been shown to be more efficient than grid or manual search [17]. In this case the values of the parameters were randomly selected from the intervals:

- (2,150) for the maximum detections parameter
- (2,150) for the maximum distance parameter
- (0,100) for the threshold parameter

The second of the optimization approaches was to use a genetic algorithm [18], [19] to select the best parameter set. Genetic algorithms are inspired by natural selection and they could be summarized as consisting of four steps: Initialization, Evaluation, Selection and Mutation/Recombination. Steps 2-4 are repeated until the termination condition is not fulfilled.

In our case in the initialization phase of genetic algorithm a population of random parameter sets is created. Afterwards, during the evaluation phase, the data is being processed by the H_DM module. Then, the resultant queue is being analyzed in order to calculate the value of fitness function – which in our case is the H_DM's function 1.

With the fitness function's values comes the selection phase when 80% of the parameters sets are discarded. Finally, in the Mutation/Recombination phase new parameter sets are created. Firstly, two random parameter sets are chosen from the remaining population. Secondly, for each parameter the new on is randomly selected from the following set:

- Value from the first parameter set
- Value from the second parameter set
- Value from the first parameter set modified by 10%

That process is repeated until the resultant population's size is equal to 40% of the starting population's size. At the end of the first iteration the population is filled with randomly-generated parameter sets in order to match the initial population size. The termination condition is set to fixed number of one hundred generations.

The next analyzed method is simulated annealing [20], [21]. It is a heuristic algorithm based on the physical process of annealing - slow transition from a high energy state to a low energy state in solid. The object being annealed firstly is heated to some high temperature so that it is possible for it to easily change its physical structure. Then it is cooling slowly in order to develop ordered crystal structure.

When it comes to the optimization algorithm firstly, there is an initialization with starting values of the temperature and a random solution. Secondly, the neighbor solution is being chosen randomly. Then, if it is better it is accepted, if worse accepted with some probability depending on the temperature parameter. Finally, the temperature parameter is being lowered. Those steps are being repeated until termination conditions are not fulfilled.

In our case at the beginning of the process one random parameter set had been chosen. Then, the new candidate parameter set is determined according to the following rules:

- new_max_detections = current_max_detections ± current_max_detections * temperature / max_temperature
- new_max_dist = current_max_dist ± current_max_dist * temperature / max_temperature
- new_threshold = current_threshold ± current_threshold * temperature / max_temperature

After that the temperature is changed to 90% of the previous value. The termination condition is defined as the temperature reaches value lower than fixed limit of 10-5 or when no changes in the H_DM function value 1 are occurring.

The last analyzed approach is the Bayesian optimization. In that case the H_DM function 1 is treated as a black-box function and is not being optimized directly itself [22], [23]. Rather than that the Gaussian process being the surrogate model of that function is being optimized with the acquisition function. At each iteration the next evaluation point in parameters space is found by finding the maximum value of the acquisition function. Then the original function is being calculated. After that the Gaussian Process is being updated with new value of H_DM function 1. The acquisition function value reflects both exploration (where the H_DM function is very uncertain) and exploitation (trying the points where the H_DM function is expected to be high). In general, the acquisition function values depends on both the Gaussian process hyperparameters and the values of the original function in previous points. In our case the Expected Improvement acquisition function is used.

V. Experiments

The experiments were conducted on each of the algorithms and tested in terms of their efficiency and effectiveness in order to find the best one for the HEIMDAL project.

We have used separate training and testing set – both represented in Table I. The training set consisted of five movies which represent the problem well. The ‘room1’ and ‘room2’ movies features two unarmed people (one per movie) walking around the room. The ‘room1-gun’ and ‘room2-gun’ movies features the same people carrying a visible gun. The ‘room5’ movie is considered to be the most difficult one as it features two people walking around the room carrying objects (e.g. a smartphone) in a way people hold a firearm. Results of evaluation of parameters sets on the training dataset are presented in Fig. 1, where points in the 3D space correspond to parameter values and the color reflects the H_DM function value (blue denoting the higher value). There is clearly no obvious maximum or general trend visible and the values are vary a lot, which makes the optimization task more ambitious.

As we focus on limiting the number of false alarms, the testing set consists of two movies ‘corridor1’ and ‘corridor4’ each featuring one hour of people walking through a corridor without any firearms. Camera placement, actors, light conditions and scene layout are different from the movies in the training set.

TABLE I. Summarization of Films in Dataset

Movie title	Num. of frames
room1-gun	2489
room2-gun	1590
room1	1391
room2	578
room5	2284
corridor1	106466
corridor4	96324

TABLE II. Comparison of Algorithms in Terms of Time and Score on the Training Set

Algorithm	Mean time of execution (s)	Average score	std.deviation of score
Random search	69.3	15.7	3.08
Genetic algorithm	2540.3	23.4	2.02
Simulated annealing	52.2	14.3	8.34
Bayesian optimization	96.8	22.2	3.14

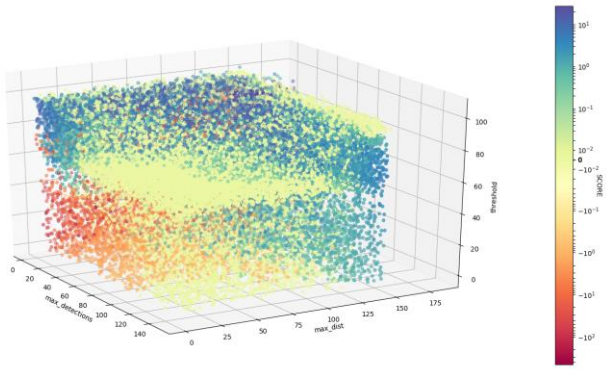

Figure 1. Results on training set.

There are a few observations worth mentioning about the results presented in Table II. The simulated annealing algorithm provides the worst result taking into consideration the average score, but its best results were competitive. On the other hand there is a genetic algorithm with the best and the most stable result obtained in the longest time. Bayesian optimization provides score comparable to genetic algorithm in 20 times shorter time.

As the evaluation on test dataset is time consuming, only Bayesian optimization and genetic algorithm were evaluated on it. Both parameter sets on both movies reported 0 detections.

VI. CONCLUSIONS

In this paper we have compared various algorithms serving hyperparameter optimization as well as their adaptation to solving the problem of choosing the best parameters for H_DM module. We opted for using the genetic algorithm as it gives, omitting time performance, the best results.

Apart from further development of the H_NN module the future researches will focus on exploration and tests upon a H_DM's function. Because of the fact that we have set up genetic algorithm parameters arbitrarily based on conducted experiments, we want try to tune them to improve the time performance of it, with preservation of its effectiveness. Gathering of a bigger and more representative dataset will serve our purpose well.

ACKNOWLEDGMENT

This work was supported by the Polish National Centre for Research and Development under the LIDER Grant (No. LIDER/354/L-6/14/NCBR/2015).

REFERENCES

[1] A. G. Howard, *et al.*, "Efficient convolutional neural networks for mobile vision applications," arXiv:1704.04861, 2017.

[2] M. Grega, *et al.*, "Application of logistic regression forbackground substitution," in *Multimedia Communications, Services and Security*, A. Dziech and A. Czyżewski, Eds., Cham: Springer International Publishing, 2017, pp. 33-46.

[3] A. Kolios, V. Mytilinou, E. Lozano-Minguez, and K. Salonitis, "A comparative study of multiple-criteria decision-making methods under stochastic inputs," *Energies*, vol. 9, no. 7, 2016.

[4] E. Triantaphyllou, "Multi-criteria decision making methods," in *Multi-criteria Decision Making Methods: A Comparative Study*, Springer, 2000, pp. 5-21.

[5] M. Grega, A. Matiolaski, P. Guzik, and M. Leszczuk, "Automated detection of firearms and knives in a CCTV image," *Sensors*, vol. 16, no. 1, 2016.

[6] S. Akcay, M. E. Kundegorski, C. G. Willcocks, and T. P. Breckon, "Using deep convolutional neural network architectures for object classification and detection within x-ray baggage security imagery," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 9, pp. 2203-2215, Sept. 2018.

[7] Z. Chen, *et al.*, "Dangerous driving behavior detection using video-extracted vehicle trajectory histograms," *Journal of Intelligent Transportation Systems*, vol. 21, no. 5, pp. 409-421, 2017.

[8] P. Lameski, E. Zdravevski, R. Mingov, and A. Kulakov, "SVM parameter tuning with grid search and its impact on reduction of model over-fitting," in *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing*, Springer, 2015, pp. 464-474.

[9] D. C. Bindushree and V. U. Rani, "A review on using various dm techniques for evaluation of performance and analysis of heart disease prediction," in *Proc. International Conference on Smart Technologies for Smart Nation*, Aug. 2017, pp. 686-690.

[10] A. Candelieri, *et al.*, "Tuning hyperparameters of a svm-based water demand forecasting system through parallel global optimization," *Computers & Operations Research*, 2018.

[11] T. Ban, M. Ohue, and Y. Akiyama, "Efficient hyperparameter optimization by using Bayesian optimization for drug-target interaction prediction," in *Proc. IEEE 7th International Conference on Computational Advances in Bio and Medical Sciences*, Oct. 2017, pp. 1-6.

[12] C. Maurice, F. Madrigal, and F. Lerasle, "Hyper-optimization tools comparison for parameter tuning applications," in *Proc. 14th IEEE International Conference on Advanced Video and Signal Based Surveillance*, Aug. 2017, pp. 1-6.

[13] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary Computation*, vol. 8, no. 2, pp. 173-195, 2000.

[14] E. Elbeltagi, T. Hegazy, and D. Grierson, "Comparison among five evolutionary-based optimization algorithms," *Advanced Engineering Informatics*, vol. 19, no. 1, pp. 43-53, 2005.

[15] N. M. A. Latiff, C. C. Tsimenidis, and B. S. Sharif, "Performance comparison of optimization algorithms for clustering in wireless sensor networks," in *Proc. IEEE International Conference on Mobile Adhoc and Sensor Systems*, Oct. 2007, pp. 1-4.

[16] D. Golovin, B. Solnik, S. Moitra, G. Kochanski, J. E. Karro, and D. Sculley. (2017). *Google Vizier: A Service for Black-Box Optimization*. [Online]. Available: http://www.kdd.org/kdd2017/papers/view/google-vizier-a-service-for-black-box-optimization

[17] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, vol. 13, pp. 281-305, 2012.

[18] T. Bäck and H. P. Schwefel, "An overview of evolutionary algorithms for parameter optimization," *Evolutionary Computation*, vol. 1, no. 1, pp. 1-23, 1993.

[19] M. Mitchell, *An Introduction to Genetic Algorithms*, MIT Press, 1998.

[20] R. A. Rutenbar, "Simulated annealing algorithms: An overview," *IEEE Circuits and Devices Magazine*, vol. 5, no. 1, pp. 1-26, Jan. 1989.

[21] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, "Optimization by simulated annealing: An experimental evaluation; part i, graph partitioning," *Operations Research*, vol. 37, no. 6, pp. 865-892, 1989.

[22] E. Brochu, V. M. Cora, and N. D. Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," arXiv preprint arXiv:1012.2599, 2010.

[23] J. Snoek, *et al.* (2012). Advances in Neural Information Processing Systems. 25. pp. 2951-2959. Curran Associates, Inc. [Online]. Available: http://papers.nips.cc/paper/4522-practical-bayesian-optimization-of-machine-learning-algorithms.pdf

Jakub Król was born in Dębica, Poland in 1996. Currently he is pursuing the engineering degree in Telecommunications on AGH University of Science and Technology in Krakow, Poland, Department of Telecommunications.

He has got 1 year experience working as a DevOps engineer.

Michal Grega was born in Krakow, Poland in 1982. He graduated from the AGH University of Science and Technology in Krakow, Poland, Department of Telecommunications. There, he obtained his PhD in Telecommunications in 2011. He specializes in image analysis and machine learning for industrial and security applications.

He is an alumni of the Tampere University of Technology, Tampere, Finland where he studied as a part of the Erasmus programme. He graduated from the Kraków University of Economics post diploma study in project management (IPMA D, IPMA R&D D certificates). He is also an alumni of University of California Berkeley, Haas School of Business, as a visiting scholar graduating from the Top500 Innovators Program on Academic Entrepreneurship. He is currently working as an Assistant Professor at the AGH university of Science and Technology, Krakow, Poland.