http://communications.science.ankara.edu.tr

Commun.Fac.Sci.Univ.Ank.Series A2-A3
Volume 65, Number 2, Pages 130-141 (2023)
DOI: 10.33769/aupse.1254312
ISSN 1303-6009 E-ISSN 2618-6462

Research Article; Received: February 21, 2023; Accepted: May 23, 2023

AN OPTIMIZED ARTIFICIAL NEURAL NETWORK FOR ESTIMATING DESIGN EFFORT OF JIGS AND FIXTURES USED IN AVIATION INDUSTRY

Umut Nazmi AKTAN[1] and Mehmet DİKMEN[2]

[1]Product Lifecycle Management Process and Method, Turkish Aerospace Company, Ankara, TÜRKİYE
[2]Computer Engineering Department, Başkent University, Ankara, TÜRKİYE

Abstract. This paper investigates the usefulness of the machine learning methods to predict the design effort of jigs and fixtures used in the aviation industry. Reaching the best possible result by determining the ideal machine learning model to obtain the best estimate and the most appropriate set of inputs and parameters forms the basis of this study. To that end, most popular machine learning models that can be used for regression are combined with various data encoding methods. The best combination is optimized as well. The results showed that an optimized Artificial Neural Network architecture with binary encoding applied to the input data can be applied satisfactorily in the aviation industry for the solution of the given problem.

1. Introduction

An accurate estimation for design effort can make a significant difference in the time and cost expectation of a project. For this reason, each new method that can be applied in the estimation of design effort has a positive effect on the schedule of the relevant projects. In aviation industry, design efforts are relatively longer than the ones in most of the other industries. Since, the parts with very few details of a typical air vehicle are interchangeable, most jigs and fixtures (tools) require producing an aero structure and the subsystems. For example, a typical two people turboprop plane requires about 5000 different jigs and fixtures to manufacture.

Keywords. Tool design effort, aviation industry, artificial neural network, machine learning.
nazmiumut@gmail.com; 0000-0002-6410-5720
mdikmen@baskent.edu.tr -Corresponding author; 0000-0002-0584-5577.

Due to the need to reduce project costs, accurate estimation of design effort is crucial, as with most problems, this need can be met with machine learning methods. To that end, this paper focuses on the estimation of tool design efforts using machine learning (ML) methods, including Artificial Neural Networks (ANN) and optimization of its parameters to achieve the best result.

2. Research

In design process, technical requirements should be clearly stated so that the design can be done in the scheduled time span. Many input factors can affect the design effort. Function, complexity, and technical requirements can be considered as examples of the input factors. The more detailed and accurate the requirements for a design are defined, the more successful the estimation of the design time can be.

The most classic method of estimating the design time is the expert opinion. For example, by directing various problems to a team of expert designers, high-impact input factors affecting project costs can be identified [1]. The method of estimating the design cost by determining the changes on the new design with the existing design data that has already been completed can also be used effectively in the change management of similar design solutions [2]. In addition, it is possible to digitize with certain parameters by examining the dependent variables that affect the design at the highest rate in the design process. By examining the matrix structure obtained in this way, it is possible to establish a relationship between the design effort and the importance of the parameters [3].

Bashir and Thomson, who have more than one study on general design effort estimation, aimed to measure the design process by establishing a relationship between product complexity and design [4]. In addition, the authors tried to estimate the design effort by using parametric simulation, regression, and analogy methods [5], [6], [7]. On the other hand, the estimation of the production effort to be made after the design is easier since the available inputs are more detailed. In this regard, a method that can predict the production time of a workpiece to be machined on a Numerical Control (NC) machine by interpreting the parameters in Computer Aided Design (CAD) programs introduced [8].

One viable way to detect design and production effort is using digital twin-based architectures. In this context, the design efforts of the proposed integrated framework design and production processes can be estimated with a Model-Based Systems Engineering structure [9].

In a study of dataless design forecasting, a contribution was made in project management and project costing by using a fast and effective method that evaluates the tacit knowledge and experience of the design teams with an analytical method [10].

According to the research where 1178 articles were examined by natural language processing method in software effort detection studies published between 1996 and 2017, it was observed that the use of machine learning methods in software effort estimation increased in the last 15 years [11]. The study showed that ML methods were generally applied on software effort or measurement, not in tool design effort. In fact, to our knowledge, ML wasn't studied for the estimation of tool design effort yet. Therefore, a throughout analysis is needed to investigate the effectiveness of ML on this problem to reduce the need of expert opinion and time. To that end, this paper presents a study that aims to estimate the design effort of the production tools in the aviation industry, by using and comparing various optimized ML methods with state-of-the-art encoding techniques applied to the input data.

3. DATASET

In this study, Tool Order data set containing various input parameters related to the tool, which was collected from the data containing the necessary information for the realization of the tool design, was used [12]. Table 1 describes the input and output parameters.

TABLE 1. Tool Order dataset.

Parameter	**Type**	**Data Type**	**Description**
TOOLCODE	Input	Categorical	<u>Tool Code</u>; describes the main function of the tool
PLANT	Input	Categorical	<u>Project Code</u>: describes the project that the tool to be used
RFO	Input	Categorical	<u>Reason for Order</u>; describes why the tool is requested
TOTYPE	Input	Categorical	<u>Tool Order Type</u>: Describes whether tool is new or to be reworked or redesigned
TOTAL	Output	Real Number	<u>Total design time</u> (hours)

The data set, which was rearranged with expert tool designer in the previous study [12], was also used in this study. The expert examined all the data and identified incorrect inputs from other designers. For example, a designer can enter wrong work order code belonging to a project that he is not working on at that time. Such inconsistencies in the data set were removed from the data content so that, they would not cause any problems that would mislead the machine learning process.

In the previous study, Linear Regression (LR), Decision Tree Regression (DTR), Support Vector Regression (SVR) with Linear kernel function, Boosted Trees Regression (BTR) and Gaussian Process Regression (GPR) methods were utilized to solve the problem [12]. In addition, following encoding techniques were examined to digitize categorical inputs: Ordinal, Binary, One Hot, Dummy, Effect (Deviation) Frequency, Mean. The best results were obtained with binary encoding and training was performed with 80% of total data which were chosen randomly. As a result of this study shown in Table 2, the GPR method gave the best results. The term RMSE represents the Root Mean Square Error.

TABLE 2. Results of the Previous Study [12].

Machine Learning Method	**RMSE**
Linear Regression	8.764
Decision Tree Regression	8.863
Support Vector Regression (Linear Kernel)	9.051
Boosted Trees (Ensemble)	8.598
Gaussian Process Regression	8.418

In this paper, parameter optimization of all these methods including an ANN model is also performed. In addition, all models are verified with 10-fold cross validation instead of separating the data set randomly.

4. METHODOLOGY

To achieve the best result for estimating tool design effort, various machine learning methods should be investigated and optimized. Therefore, a framework specified in Figure 1 is proposed. Since, other encoding techniques for the methods given Table 2 were already examined in the previous study [12], only the best encoding technique for ANN is investigated in this paper.

ANN, which is a set of mathematical models inspired by nature, can be defined as a method of arranging the parameters of a set of nonlinear combined functions with input and output sets. The ANN architecture in this study can be summarized in Figure 2, where n and m are the number of neurons in input and hidden layer respectively, w and v are the real number weights between 0 and 1 [13]. Since, there is just one value to be estimated (i.e., the total design time), ANN contains a single output neuron. Also, bj values are used as initial arbitrary constants (i.e., biases) to shift the regression functions. fj's represent the activation functions which are used to calculate the output (v_{fj}) of each neuron using the following sigmoid function.

$$f_j(x) = \begin{cases} g_j(x), & if \ \frac{1}{1+e^{-g_j(x)}} > 0 \\ 0, & otherwise \end{cases}$$

where $g_j(x)$ represent the net input of a neuron calculated as:

$$g_j(x) \text{ is } g_j(x) = \sum_{i=1}^{n} x_i w_{x_i,j} + b_j$$

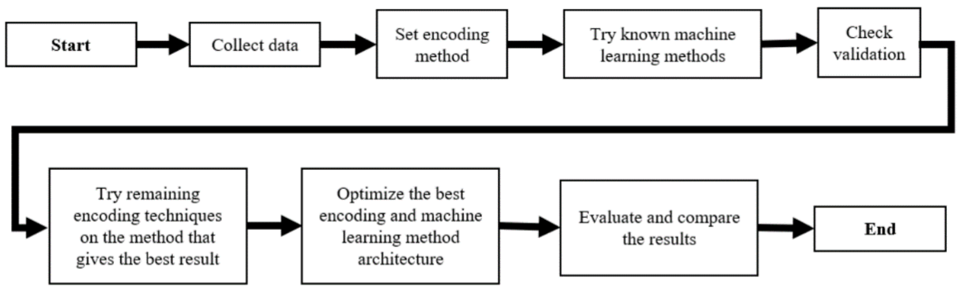

FIGURE 1. Framework to determine the best encoding and machine learning method.

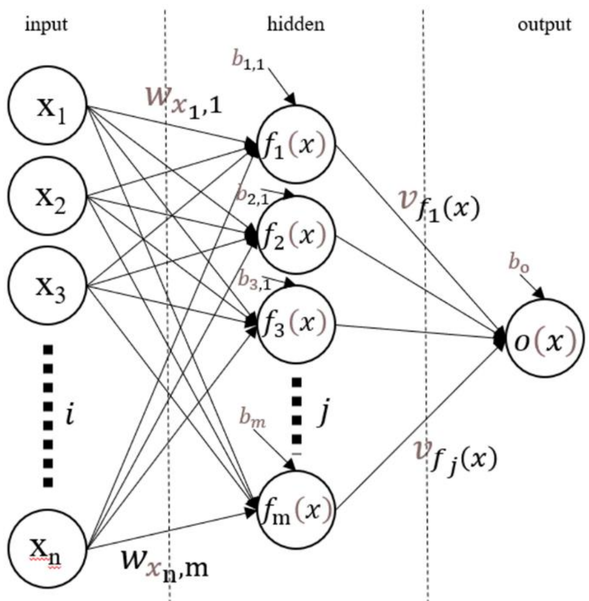

FIGURE 2. The initial ANN model diagram.

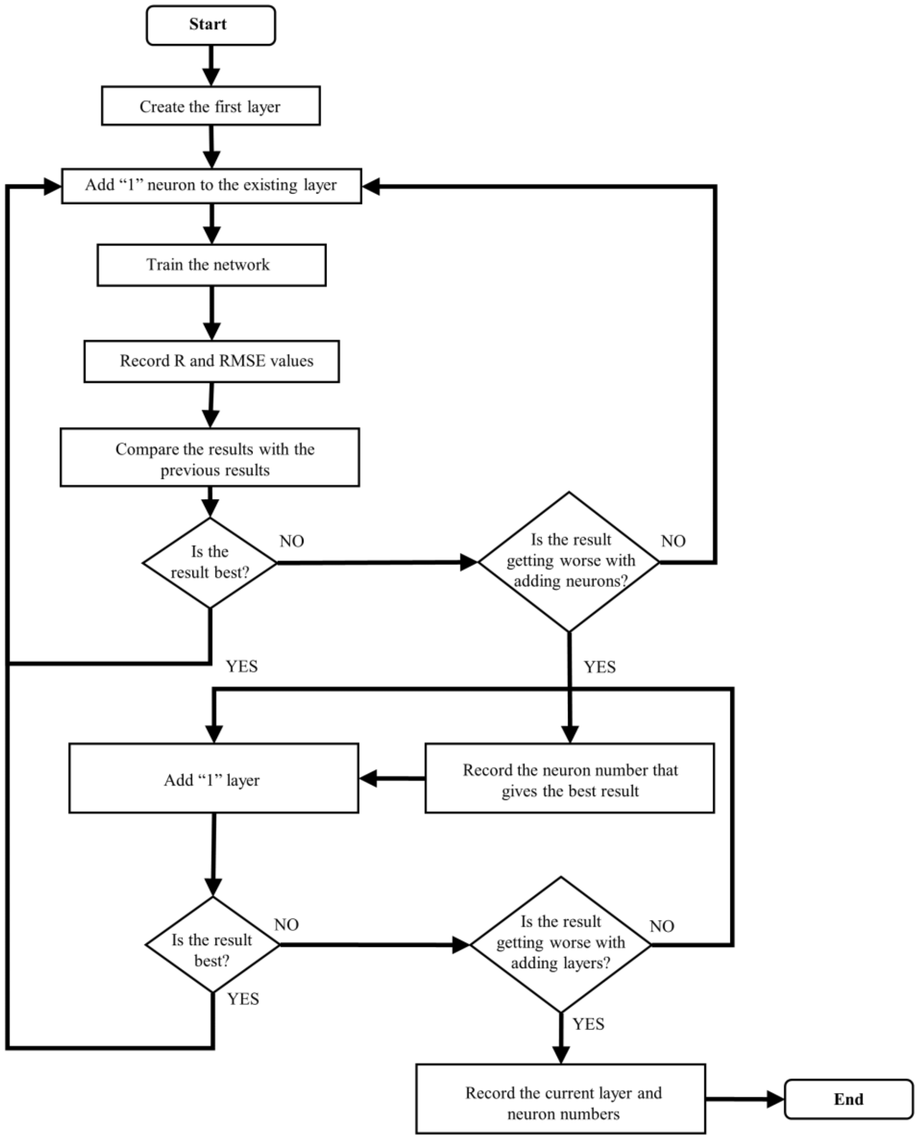

FIGURE 3. Suggested workflow for optimizing the ANN.

Training the ANN model was done by using Levenberg-Marquardt algorithm presented by Levenberg and Kenneth, which provides a fast convergence without computing the exact Hessian matrix [14]. In addition, early stopping technique was employed to avoid the overfitting problem. For this purpose, 10% of the training data was randomly selected as the validation dataset and the prediction error on this dataset was observed. If no improvement is observed on validation error after 10 iterations, the training is stopped due to overfitting, and the best model weights were restored.

5. Parameter Optimization of Machine Learning Models

Each method except Linear Regression (LR) has parameters to optimize which are addressed in next paragraphs. The optimization procedure is limited to 500 iterations or 24 hours of computation time. For each parameter mentioned below, possible values are iterated. Only the parameter-value list that gave the best results is presented for each machine learning model.

The decisive parameter for optimizing DTR is evaluated as the range of each tree leaf. Additionally, surrogate decision splits and maximum surrogates per node were included in this evaluation process.

BTR requires four parameters Minimum leaf size, number of learners, number of predictors to sample and log scaled learning rate to be variated during optimization.

In the SVMR optimization process, 4 kernel functions were examined, which are Gaussian, Linear, Quadratic, and Cubic. The scales of these functions, which are log scaled, were also evaluated. Moreover, box constraint and epsilon (Ɛ) values were considered.

In GPR optimization, 3 basis functions (Zero, Constant, and Linear) and 10 kernel functions were cycled (Non-isotropic Rational Quadratic, Isotropic Rational Quadratic, Non-isotropic Squared Exponential, Isotropic Squared Exponential, Non-isotropic Matern (Genton,2001) 5/2, Isotropic Matern 5/2, Non-isotropic Matern 3/2, Isotropic Matern 3/2, Non-isotropic Exponential, Isotropic Exponential).

ANN optimization involves the decision of how many hidden layers there should be and how many neurons each layer should have. Although there are techniques, such as using Genetic Algorithm to find the optimal parameter combination [15], trial and error technique is preferred in this study for its simplicity. It must be noted that the number of input neurons could be more than the number of inputs (i.e., 4) to the problem, since the exact number of inputs are increased after applying an encoding technique. Since, the input size is not same for all encoding techniques, the number of input neurons of the ANN varies depending on the encoding technique used. To conclude, to optimize the ANN structure, the workflow proposed in Figure 3 is presented. This procedure basically describes that layer size and neuron size (i.e., the number of neurons in a layer) are increased one by one until the best RMSE value is achieved. If, in any stage, adding a new neuron starts to worsen the result with a tolerance of 10 additions, no new neuron is added anymore and the number of neurons that gave the best RMSE so far is accepted for that layer. If the inclusion of that layer has improved the result, then the procedure continues by adding another layer. Otherwise, the optimization stops and accepts the layer-neuron combination that gave the lowest RMSE value so far.

6. Results and Discussion

The results of the ANN optimization procedure in Figure 3 are presented in Table 3, where rows represent the encoding techniques, and the columns correspond to the structure of the ANN. These results were obtained for each encoding technique within the scope of the proposed framework in Figure 1. Examining Table 3, the best result was obtained using binary encoding technique on an ANN with 2 hidden layers. The layers are consisted of 206 and 95 neurons, respectively. Since binary encoding produced 23 features, this ANN architecture has 23 input neurons. In general, the increase in the number of layers in ANN architecture had a positive effect on the result and having more than 3 hidden layers decreased the performance. On the other hand, the one hot encoding method quickly gave worse results with few neurons in the 2nd hidden layer. This can be caused by the fact that this technique generated the most features (i.e., 189) which increased the dimensionality of data and make it harder to generalize.

Table 3. ANN optimization results.

Encoding Method	**Layer - 1**		**Layer - 2**		**Layer - 3**		**Layer - 4**	
	#Neurons	**RMSE**	**#Neurons**	**RMSE**	**#Neurons**	**RMSE**	**#Neurons**	**RMSE**
Ordinal	337	8.672	35	8.507	17	8.493[1]	60	8.534[2]
Binary	206	7.523	95	7.184[3]	87	7.307[2]	-	-
One Hot	4	8.052[1]	11	8.226[2]	-	-	-	-
Dummy	202	7.999	152	7.972[1]	16	8.058[2]	-	-
Effect	210	8.014[1]	68	8.106[2]	-	-	-	-
Frequency	168	8.543	126	8.455	54	8.367[1]	33	8.429[2]
Mean	119	8.170	323	7.974[1]	330	8.239[2]	-	-

[1] Best result for a particular encoding method.
[2] Optimization got worse so stopped.
[3] Optimal encoding and ANN architecture.

TABLE 4. Optimized results of all machine learning models.

Machine Learning Model	**RMSE**
Linear Regression	8.764
Decision Tree Regression	8.827
Support Vector Regression (Quadratic Kernel)	8.812
Boosted Trees (Ensemble)	8.571
Gaussian Process Regression	8.401
Artificial Neural Network	7.184

Optimized results of all machine learning models are presented in Table 4, where optimal value of each parameter is given in Table 5. Examining Table 4 together with Table 3, the results of all models were slightly improved after their parameter optimization. Despite those improvements, the optimized ANN architecture unquestionably obtained the best overall result. An interesting result in Table 4 shows that even a simple model like Linear Regression can achieve comparable results with other models. This might originate from the low representation capability of the current parameters. Unarguably, inclusion of additional features will represent the data better, thus more accurate results could be obtained.

TABLE 5. Optimal parameter values for each machine learning model.

DTR		**SVR**		**BTR**		**GPR**	
min leaf size	24	kernel function	quadratic	min leaf size	11	basis function	constant
surrogate decision split	off			number of learners	428	kernel function	isotropic squared exponential
				number of predictors to sample	8		
				learning rate	adaptive		

The business intelligence system of the organization that owns the data set performs project follow-ups daily. The tool design schedule is also included in this business intelligence system. When looking at the process on a long scale, for instance, in tool design project planning, it is preferred to use "day" units instead of "hour" units; and in some cases, "month" units in efforts calculation since the project calendars are usually expressed in years. Therefore, missing the correct design time by 10-15 hours is in fact not a bad estimation since the estimation error is still less than 1 day. In other words, if design times were given as days instead of hours, RMSE values would be smaller than the ones in Table 4. For this reason, model performances should also be assessed in an alternative way. To that end, following rule of thumb is used: in aviation industry, estimates can be considered as correct if the time allocated to tool design engineering does not exceed 20% of the total duration of the project. Table 6 presents the estimation accuracy of each method when such an evaluation is done, where an estimation is assumed to be correct if its difference to actual design time do not exceed 20% of the actual design time. As a result, it is observed that ANN proved superior to other ML models by correctly estimating the 85.64% of the total data.

TABLE 6. Estimation accuracies of all machine learning models.

Machine Learning Model	**Accuracy**
Linear Regression	0.7704
Decision Tree Regression	0.7665
Support Vector Regression (Quadratic Kernel)	0.7525
Boosted Trees (Ensemble)	0.7890
Gaussian Process Regression	0.8049
Artificial Neural Network	0.8564

7. CONCLUSIONS

In this study, an optimized Artificial Neural Network architecture is proposed for the estimation of the design effort of production tools used in the aviation industry. This approach can also be adapted to many other design processes if requirements are well defined and prior effort data is available.

In the beginning of this study all machine learning methods applied on the dataset gave comparable results. However, ANN and binary encoding gave the most successful results due to the better representation capability of binary encoding and parameter optimization.

When the relationship between error rates and tool attributes are examined; in production projects where the product design was completed, known as build to print, the estimation performance has been found to be much better than the estimation of development projects. Comparing the relationship between the characteristic of the design and the error level; it has been seen that design studies involving generic activity are predicted with less error. For instance, preparing a periodic measurement document for a tool was much better predicted than designing a tool from scratch. When the tool code and error rate were examined, no significant relationship was found between them. For this reason, it was evaluated that the main function of a tool alone was not decisive.

It is also thought that high dimensionality of the input data (i.e., more input neurons in input layer in ANN) might also degrade the performance of the proposed system. Therefore, dimension reduction techniques, such as Principal Component Analysis and Autoencoder methods, may also be considered as a future work to assess whether this is the case or not.

Author Contribution Statements Authors are equally contributed to the paper. All authors read and approved the final copy of the manuscript.

Declaration of Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement This study was supported by Turkish Aerospace Company.

References

[1] Benedetto, H., Vieira, D., Proposed framework for estimating effort in design projects, *Int. J. Manag. Proj. Bus.*, 11 (2) (2018), 257-274, https://doi.org/10.1108/IJMPB-03-2017-0022.

[2] Martin, M. V., Ishii, K., Design for variety: a methodology for understanding the costs of product proliferation, *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 97607 (1996), https://doi.org/10.1115/96-DETC/DTM-1610.

[3] Dentsoras, A. J., Information generation during design: information importance and design effort, *AI EDAM*, 19 (1) (2005), 19-32, https://doi.org/10.1017/S089006040505002X.

[4] Bashir, H. A., Thomson, V., Estimating design complexity, *J. Eng. Des.*, 10 (3) (1999), 247-257, https://doi.org/10.1080/095448299261317.

[5] Bashir, H. A., Thomson, V., Estimating design effort for GE hydro projects, *Comput. Ind. Eng.*, 46 (2) (2004), 195-204, https://doi.org/10.1016/j.cie.2003.12.005.

[6] Bashir, H. A., Modeling of development time for hydroelectric generators using factor and multiple regression analyses, *Int. J. Proj. Manag.*, 26 (4) (2008), 457-464, https://doi.org/10.1016/j.ijproman.2007.08.006.

[7] Bashir, H. A., Thomson, V., An analogy-based model for estimating design effort. *Des. Stud.*, 22 (2) (2001), 157-167, https://doi.org/10.1016/S0142-694X(00)00015-6.

[8] Ou-Yang, C., Lin, T., Developing an integrated framework for feature-based early

manufacturing cost estimation, *Int. J. Adv. Manuf. Technol.*, 13 (9) (1997), 618-629, https://doi.org/10.1007/BF01350820.

[9] Wang, H., Li, H., Wen, X., Luo, G., Unified modeling for digital twin of a knowledge-based system design, *Robot. Comput. Integr. Manuf.*, 68 (2021), 102074, https://doi.org/10.1016/j.rcim.2020.102074.

[10] Thomson, A., Hird, A., Estimating design effort needs of product design projects using captured expert knowledge–a proposed method, *Proceedings of the Design Society*, 1 (2021), 1391-1400, https://doi.org/10.1017/pds.2021.139.

[11] Sehra, S. K., Brar, Y. S., Kaur, N., Sehra, S. S., Research patterns and trends in software effort estimation, *Inf. Softw. Technol.*, 91 (2017), 1-21, https://doi.org/10.1016/j.infsof.2017.06.002.

[12] Aktan, U., Dikmen, M., Estimation of design effort of jigs and fixtures used in aviation industry by machine learning methods, *2nd International Eurasian Conference on Science, Engineering and Technology*, (2020), 500-505.

[13] Genton, M. G., Classes of kernels for machine learning: a statistics perspective, *J. Mach. Learn. Res.*, 2 (2001), 299-312.

[14] Levenberg, K., A method for the solution of certain non-linear problems in least squares, *Q. Appl. Math.*, 2 (2) (1944), 164-168.

[15] Soylu, K., Güzel, M. S., Emek Soylu B., Bostanci, G. E., Genetic hyperparameter optimization library development and its application on plant disease detection problem, *28th Signal Processing and Communications Applications Conference (SIU)*, (2020), 1-4, https://doi.org/10.1109/SIU49456.2020.9302246.