IOP Conf. Series: Materials Science and Engineering **245** (2017) 062030 doi:10.1088/1757-899X/245/6/062030

Artificial Neural Networks as an Architectural Design Tool- Generating New Detail Forms Based On the Roman Corinthian Order Capital

Kacper Radziszewski[1]

[1] Gdansk University of Technology, Faculty of Architecture, Narutowicza 11/12, 80-233 Gdansk, Poland

kacperradziszewski@outlook.com

Abstract. The following paper presents the results of the research in the field of the machine learning, investigating the scope of application of the artificial neural networks algorithms as a tool in architectural design. The computational experiment was held using the backward propagation of errors method of training the artificial neural network, which was trained based on the geometry of the details of the Roman Corinthian order capital. During the experiment, as an input training data set, five local geometry parameters combined has given the best results: Theta, Pi, Rho in spherical coordinate system based on the capital volume centroid, followed by Z value of the Cartesian coordinate system and a distance from vertical planes created based on the capital symmetry. Additionally during the experiment, artificial neural network hidden layers optimal count and structure was found, giving results of the error below 0.2% for the mentioned before input parameters. Once successfully trained artificial network, was able to mimic the details composition on any other geometry type given. Despite of calculating the transformed geometry locally and separately for each of the thousands of surface points, system could create visually attractive and diverse, complex patterns. Designed tool, based on the supervised learning method of machine learning, gives possibility of generating new architectural forms- free of the designer's imagination bounds. Implementing the infinitely broad computational methods of machine learning, or Artificial Intelligence in general, not only could accelerate and simplify the design process, but give an opportunity to explore never seen before, unpredictable forms or everyday architectural practice solutions.

1.Introduction

In the recent years Machine Learning [1] became one of the most promising method of computing, influenced by the access and growth of the Big Data[2], decrease of the cost per computing operation[3] and anticipated constant increase in the available computing power[4]. Machine learning application[5] can be found on programming everyday basis, yet in architectural design is generally limited to an image processing [6, 7, 8].

The presented experiment investigates the process, possibilities and implementation of the Artificial Neural Networks, [9] (ANN) trained based on the spatial properties of 3-dimensional shapes of the Roman Corinthian order capital, being the step for further research in the field of machine learning and architectural design.

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

2.Data selection

2.1. Neural Network Training Set

The ANN training data set consists of one 3d model of Roman Corinthian order capital [10], being divided into samples regular UV surface division coordinates. Any other capital given as the training set will result in differently trained ANN. Another possible approach is using more than one 3d model as the one set of input data.

2.2. Input Vectors

Based on the visual analysis of the Roman Corinthian order capital in the first phase ten Input Vectors were selected to be verified on the influence on the minimizing the Mean Squared Error [11] (MSE) during the ANN learning process.

The set of test input:

- (x,y,z) coordinates in cartesian system of the sample,
- spherical coordinates (r, θ, φ) with a coordinate origin in a volumetric centre of the capital,
- sampled surface local normal vector deviation from the global coordinate system,
- sampled surface local curvature,
- number of neighbourhood samples- detecting the boundaries of the capital,
- distance from xz and yz plane with origin in a volumetric centre of the capital

During the trial and error method of problem solving [12] based on the minimizing the MSE value, simple Artificial Neural Network was set (later being adjusted based on the selected input vectors). The lowest MSE value was found using six of the presented, possible, input vectors:

- z coordinate in cartesian system of the sample
- spherical coordinates (r, θ, φ) with a coordinate origin in a volumetric centre of the capital,
- sampled surface local normal vector deviation from the global coordinate system,
- distance from xz and yz plane with origin in a volumetric centre of the capital
-

2.3. Output Vector

The algorithm output vector is created as a displacement of the capital's details 3d form calculated from the basic surface below them. Each of the basic surface points displacement were calculated based the sampled surface point normal vector intersection with the 3d Corinthian order capital detailed model (figure 1).

Given distances were remapped into domain of 0.0 to 1.0, creating displacement map, with each of the values being a single output vector for a matching training set. Presented approach simplifies the idea of ornament generating using machine learning algorithm, as it doesn't take into consideration the exact 3-dimensional composition details, but rather 1-dimensional displacement.

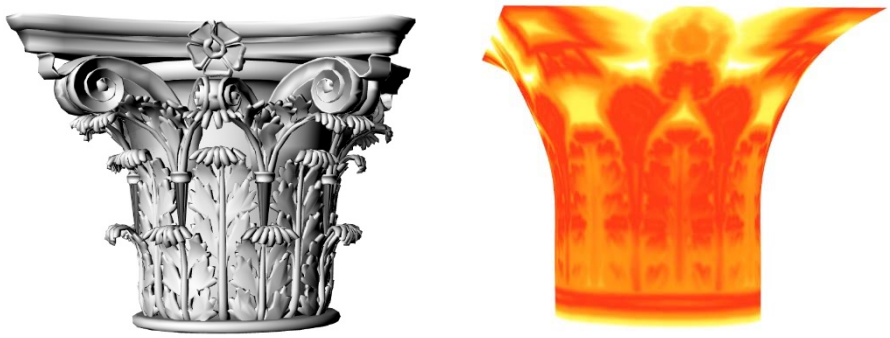
Calculated set of the ANN Input Vectors along with the chosen Output vectors were used as a base for adjusting the optimized ANN layers count, activation function, neurons count, learning rate, [13] value and samples sized used during ANN supervised learning[14] process.

3.Artificial Neural Network architecture

3.1. Artificial Neural Network type selection

The purpose of the presented system was to learn the correlation between given base surface selected parameters (Training Set Input Vectors) and sampled displacement values (Training Set Output Vector). The learning from examples[15] machine learning approach was selected based on the types of data used in the experiment and the aim of exploration the concept of teaching the program to make design decision based on the given human- made forms. The written algorithm uses the ANN

computational approach with back-propagation of error learning procedure[16]. The selected system allows the algorithm to adjust the weights between Artificial Neurons, responsible for computing the data, inside the Network based on the difference between the current ANN output value and the value being given as a target- training set output vector. During the learning procedure back-propagation of errors algorithm adjusts the system, which once programmed, is able to compute any other given input vectors set- mimicking the human learners, who are able to learn from incomplete, contradictory information, even in the absence of relevant background knowledge[17].



a) b)

Figure 1. a) Corinthian Order Capital 3d model; b) Generated displacement map;

3.2. Artificial Neural Network layers structure

The essential part of designing the ANNs is setting the proper structure of the network. Several values have to be set in order to begin the training of the final ANN: number of hidden layers, count of the neurons in each of the hidden layers, activating function for each of the hidden layers and one constant learning rate of the network. With the basic assumption of having from 1 to 3 hidden layers, where each of them could have 3-15 neurons, each of the layers could use 1 of 5 activating functions(Sigmoid, Linear, Logarithmic, Sine, TanH) and learning rate in a domain from 0.1 to 1.0, there could be 24 716 250 possible combinations.
Based on the large number of possible ANN architectures there has been used an Evolutionary Algorithm [18] (EA) in order to optimize the ANN structure. The goal of the EA was to minimize the ANN Mean Square Error based on the set of 1600 training samples (medium set has been chosen due to the long computing time) by optimizing the ANN parameters: number of hidden layers, hidden layers neurons, activating function and learning rate. After 53 populations computed, initializing the algorithm with a random structure ANN structure, the optimal parameters have been found: 3 hidden layers (12,10,7), sigmoid activating function for each of the layers and 0.17 learning rate.

3.3 Input Data sample selection

Based on the previously set ANN architecture, Corinthian order capital sampling density was set by comparing it with the trained ANN Mean Square Error. Each of the sampling size is a capital base surface UV division count, started with the UV division value of 10(100 samples) and pausing at value of 170(28 900 samples) resulting in the final trained back-propagation of errors ANN of 0.000774 Mean Square Error (figure 2).

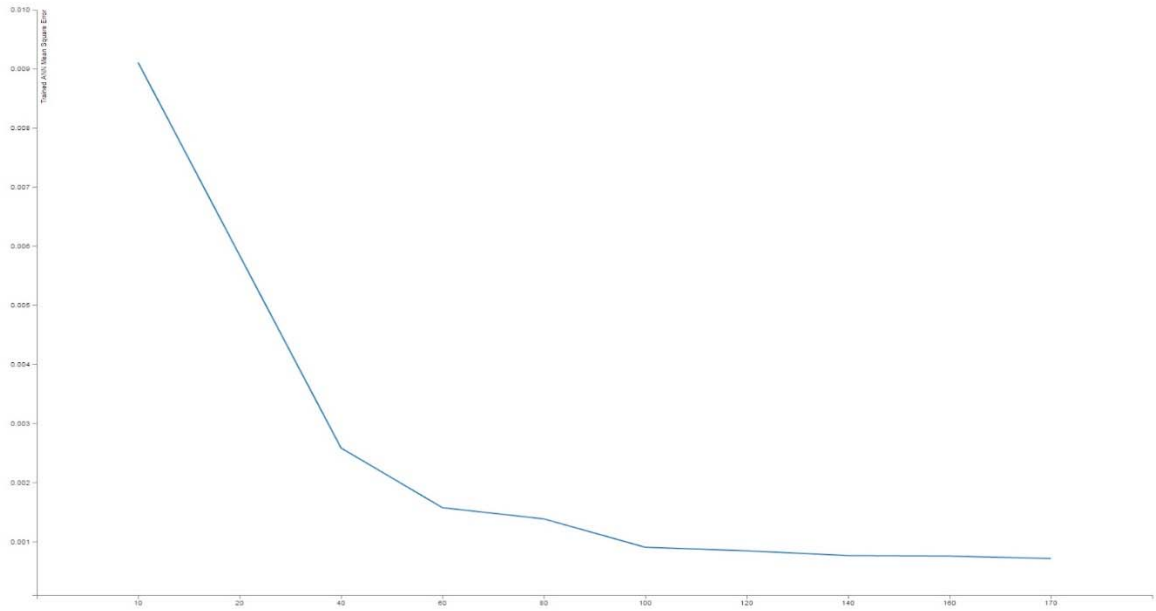


Figure 2. Surface UV division sample and MSE comparison

3.4. Final trained Artificial Neural Network structure

Back propagation of errors artificial neural network (figure 3):

- 6 input vectors
- 1 output vector
- 3 hidden layers:
  12 hidden neurons (sigmoid activating functions [19])
  10 hidden neurons (sigmoid activating functions)
  7 hidden neurons (sigmoid activating functions)
- learning rate: 0.17
- training set input vectors sets count = 28 900
- Artificial Neural Network Mean Square Error = 0.000774

4. Results

4.1. Test Data Set

Further step, before using the ANN in a design process, was examining the difference between actual displacement values and displacement values predicted by the ANN, based on the other 3d model of the Corinthian order capital. The test data set shown that the average difference is minor, below 6% error, comparing 3d models displacement(x-axis) and displacement predicted by ANN(y-axis) (figure 4). The graph shows that the ANN has not predicted values of displacement lower than 0.01 and higher than 0.95. Experiment shows that more extreme 3d model displacement is, the higher error can be expected. Such data structure can be explained by generalization feature [20] of the ANN.

4.2. Generating Capital Patterns

Designed digital tool allowed to generate displacement maps from simple surface (figure 5a), which applied to the source surface creates modified shape with a visible, organized and ornament-like spatial structure (figure 5b). As the ANN input data was collected from a single Corinthian order capital, second ANN, using the same network architecture, was trained with a different capital, examining its universal usability and the difference in the generated forms (figure 5c). The ANN can

be trained with a different, single given examples as well with a group of objects representing the same style and features, what leads to creating universal trained ANN, yet computing such a large data set will require far more computing power available.

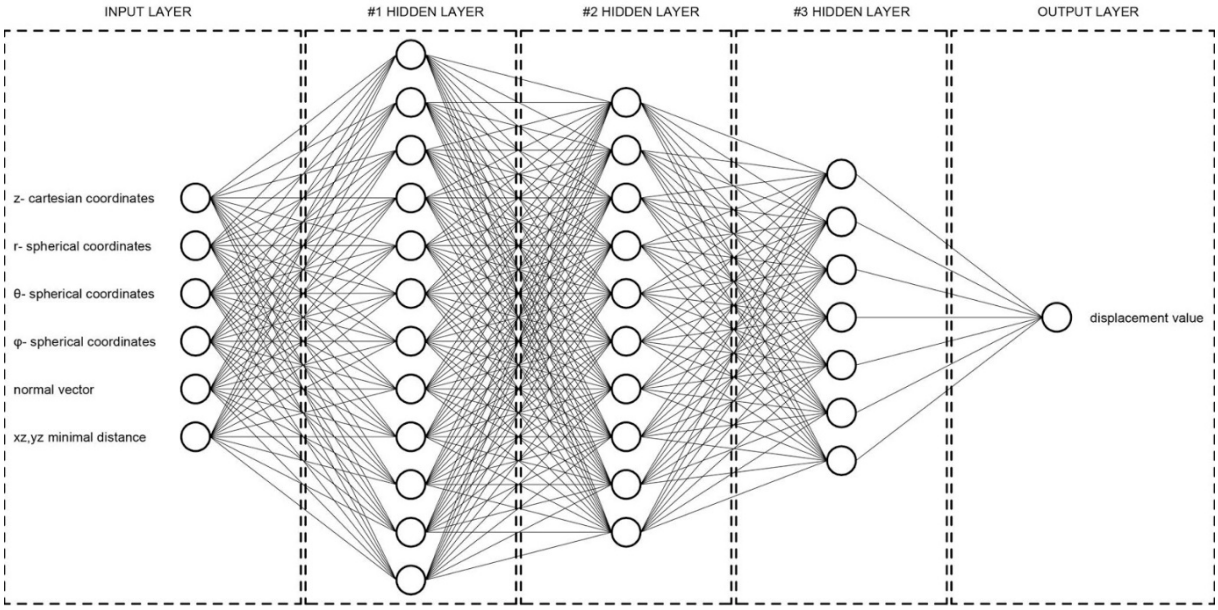


Figure 3. Trained Artificial Neural Network architecture;

Once properly trained ANN allowed applying it to any given surface or set of combined surfaces. Automated process of generating base surfaces created a set of possible capitals [21] what can be used as an catalogue of concept forms. Augmenting the design process by generating various results, allows supporting the designers creativity process, by proposing different solution. In addition to generating the capitals, each of the solutions can be used as a step in a further design process. Computed and selected information can be merged, used as an additional training set data, resulting in a more unpredictable outputs. Another approach of creative ANN use is additional transforming the output data, by its manipulation- multiplying by mathematical functions (figure 6) or using twice as input ANN input data, possibilities are vast, depended on author's creativity. ANN can work as problem solving tool, as well a creative method, augmenting the design process.

4.3. Universal application of trained Artificial Neural Network

Each of the trained ANN if provided the input data, computes it, providing the output, even if the given data is far different from the training set. Five input parameters: Theta, Pi, Rho in spherical coordinate system based on the capital volume centroid, followed by Z value of the cartesian coordinate system and a distance from vertical planes created based on the capital symmetry used during the training process are common for any other type of geometry given. Feature enables using any given sampled 3- dimensional form as the ANN input data, generating its displacement values and transforming it (figure 7).

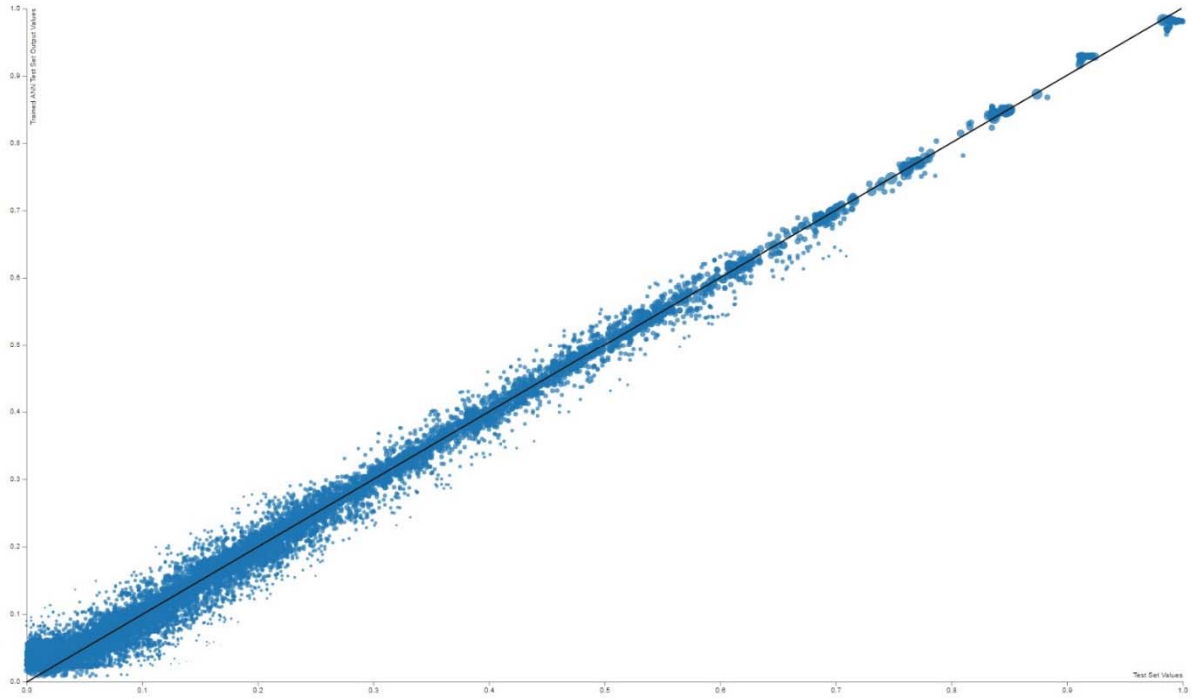
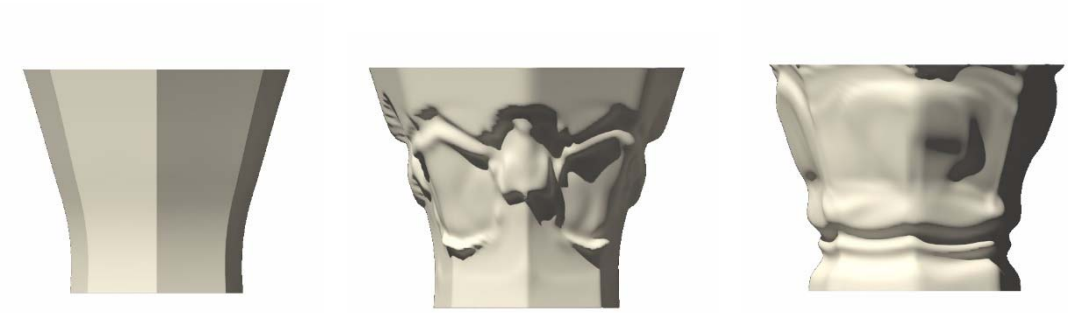
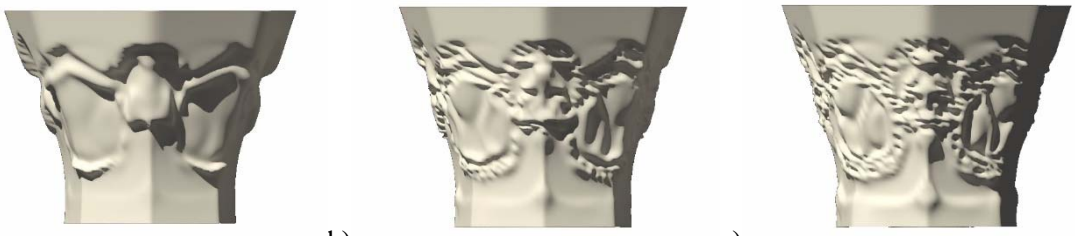


Figure 4. Test Data Set output value comparison



a) b) c)

Figure 5. (a) Base surface for ANN input collection; (b) Surface applied displacement generated based on the #1 ANN; (c) Surface applied displacement generated based on the #2 ANN;



a) b) c)

Figure 6. (a) Surface applied displacement generated based on the #1 ANN; (b) Displacement with sine function applied; (b) Displacement with 3*sine function applied;

The presented approach could be used as well as a tool in architectural 3- dimensional design, yet generating only a visually attractive forms, not solving the gap between the available digital tools and a wide range of possibilities of using ANN for full-scale, multi-layered architectural design.

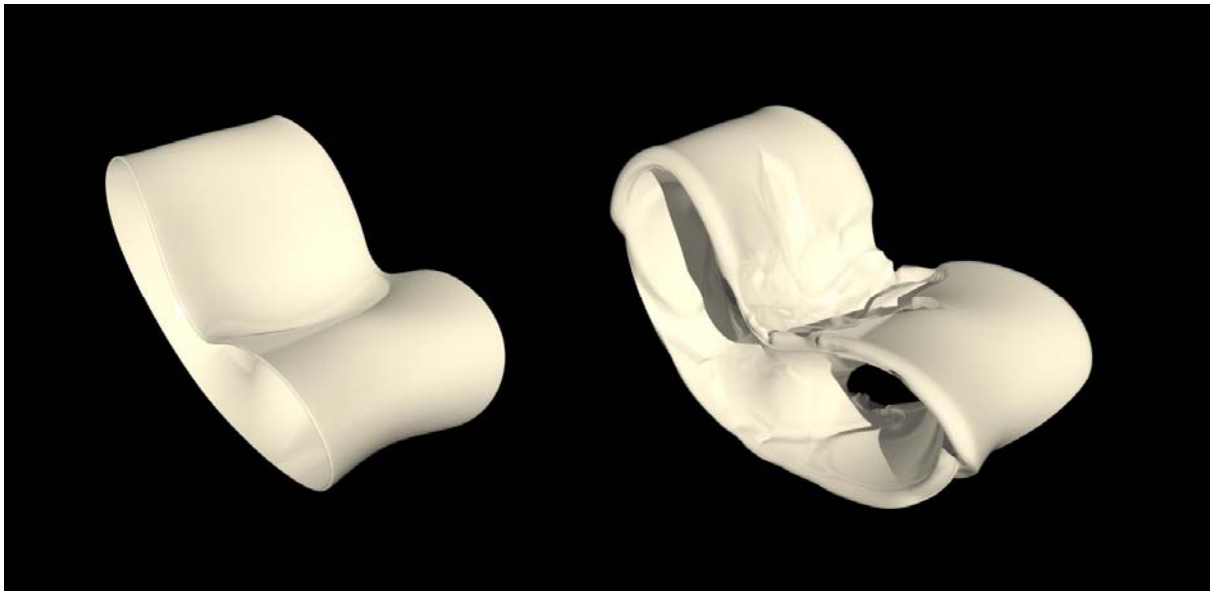


Figure 7. (a) 3d model of sofa as an ANN input data set; (b) application of an ANN trained based on the Corinthian order capital on a sofa 3d model;

5.Further prospect and conclusions

The further experiments should take into consideration using the capital details as a whole component being oriented on the provided geometry by the trained ANN, in a place of using the displacement map as an output.

Supervised learning method of Artificial Neural Networks enables creating fast computing tool for problems solving being impossible to program, because of the complexity of data dependencies. As well as problem solving tool, ANN can be used as the augmentation of the designer's creativity process.

Machine learning techniques provide many methods which can implemented currently to the common architectural practice because of the limitations of the parametric design software, the growth in the popularization of programming skills among the architects and a need of automatization of the architectural design.

References

[1] A. L. Samuel, "Some studies in machine learning using the game of checkers," IBM Journal of research and development., vol. 3, No. 3, 1959.

[2] D. Agrwal, P. Bernstein, E. Bertino, S. Davidson, U. Dayal, M. Franklin..., "Challenges and Opportunities with Big Data: A white paper prepared for the Computing Community Consortium committee of the Computing Research Association," 2012.

[3] D. Dorn, "The Rise of the Machines: how computers have changed work," UBS Center Public Papers 4, 2015.

[4] G. E. Moore, "Cramming more components onto integrated circuits," Electronics, vol. 38, Number 8, pp. 114-117, 1965.

[5] S. Das, A. Dey, A. Pal, N. Roy, "Application of Artificial Intelligence in Machine Learning: Review and Prospect," International Journal of Computer Applications (0975-8887), vol. 115, No. 9, pp. 31-41, 2015.

[6] J. Silvestre, Y. Ikeda, F. Guéna, "Artificial imagination of architecture with deep convolutional neural network "Laissez-faire": Loss of control in the esquisse phase," CAADRIA 2016, 21st International Conference on Computer-Aided Architectural Design Research in Asia - Living Systems

and Micro-Utopias: Towards Continuous Designing, pp. 881-890, 2016.
[7] P. Isola, J. Zhu, T. Zhou, A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," Barkeley AI Research (BAIR) Laboratory, University of California, 2016.
[8] https://opendot.github.io/ml4a-invisible-cities/ (access 03/2017).
[9] N. K. Kasabov, "Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering," Bradford Book The MIT Press Cambridge, pp. 251-259, 1996.
[10] http://www.sharecg.com/v/68677/related/5/3d-model/corinthian-column (access 03/2017).
[11] E. L. Lehmann, EG. Casella, "Theory of Point Estimation (2nd ed.), Springer, 1998.
[12] D. T. Campbell, "Blind Variation and Selective Retention in Creative Thought as in Other Knowledge Process," Evolutionary Epistemolog, Rationality, and the Sociology of Knowledge, Open Court Publishing Company, 1993.
[13] I. C. Peace, A. O. Uzoma, S. A. Ita, "Effect of Learning Rate on Artificial Neural Network in Machine Learning," International Journal of Engineering Research & Technology, vol. 4, 2015.
[14] N. K. Kasabov, "Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering," Bradford Book The MIT Press Cambridge, pp. 260-269, 1996.
[15] R. S. Michalski, J. G. Carbonell, T.M. Mitchell, "Machine Learning, An Artificial Intelligence Approach," Springer-Verlag Berlin Heidelberg, pp. 9-10, 1983.
[16] D. E. Rumelhart, G. E. Hinton, R. J. Williams, "Learning representations by back-propagating errors," Nature, vol. 323, pp. 533-536, 1986.
[17] D. Fisher, M. Pazzani, "Concept Formation: Knowledge and Expreience in Unsupervised Learning," Morgan Kaufmann Publishers, pp. 3-44, 1991.
[18] X. Yu, M. Gen, "Introduction to Evolutrionary Algoritms," Springer-Verlag London Limited, pp. 3-9, 2010.
[19] F. Rattay, "Analysis of Models for External Stimulation of Axons," IEEE Transactions on Biomedical Engineering, vol. 10, pp. 974-977, 1986.
[20] S. Urolagin, K.V. Prema, N.V. Subba Reddy, "Generalization Capability of Artificial Neural Network Incorporated with Pruning Method," Thilagam P.S., Pais A.R., Chandrasekaran K., Balakrishnan N. (eds) Advanced Computing, Networking and Security. ADCONS 2011. Lecture Notes in Computer Science, Springer, vol. 7135, 2012.
[21] https://drive.google.com/file/d/0B_ZFRGnTFhuhVS1pZDhPTHNOQTg/view (access 03/2017)