

Deep Galaxy V2: Robust Deep Convolutional Neural Networks for Galaxy Morphology Classifications

Nour Eldeen M. Khalifa^{1,3}
nourmahmoud@cu.edu.eg

Mohamed Hamed N. Taha^{1,3}
mnasrtaha@cu.edu.eg

Aboul Ella Hassanien^{1,3}
aboitcairo@cu.edu.eg

I. M. Selim²
i_selim@yahoo.com

¹Information Technology Department, Faculty of Computers and Information, Cairo University, Giza, Egypt

²National Research Institute of Astronomy and Geophysics, Cairo, Egypt

³Scientific Research Group in Egypt (SRGE) <http://www.egyptscience.net>

Abstract— This paper is an extended version of “Deep Galaxy: Classification of Galaxies based on Deep Convolutional Neural Networks”. In this paper, a robust deep convolutional neural network architecture for galaxy morphology classification is presented. A galaxy can be classified based on its features into one of three categories (Elliptical, Spiral, or Irregular) according to the Hubble galaxy morphology classification from 1926. The proposed convolutional neural network architecture consists of 8 layers, including one main convolutional layer for feature extraction with 96 filters and two principle fully connected layers for classification. The architecture is trained over 4238 images and achieved a 97.772% testing accuracy. In this version, “Deep Galaxy V2”, an augmentation process is applied to the training data to overcome the overfitting problem and make the proposed architecture more robust and immune to memorizing the training data. A comparative result is present, and the testing accuracy was compared with those of other related works. The proposed architecture outperformed the other related works in terms of its testing accuracy.

Keywords— galaxies classification, Deep Convolutional Neural Networks.

I. INTRODUCTION

Finding clues about the origin and the evolution of the universe remains a considerable challenge for astrophysicists. Galaxy classification is used to help astrophysicists face this challenge and is done using huge databases of information, which help astrophysicists test theories and find new conclusions to explain the physics of the processes governing galaxies, star formation, and the nature of the universe [1].

The increase in the sizes of telescopes and the advent of the CCD camera has generated extremely large image-based datasets. A huge amount of effort and time is required for any manual classification and analysis of these datasets. Galaxy morphology classification is based on images and spectra and was considered a long-term goal for astrophysicists and to be used in educational purposes. However, the complex natures of galaxies and the qualities of the images have made the classification of galaxies challenging and inaccurate [2].

The galaxy morphology classification system helps astronomers group galaxies by their visual shapes. The Hubble galaxy sequence is considered one of the most used schemes of galaxy morphological classification.

The Hubble galaxy classification scheme (Figure 1) divides galaxies into 3 categories according to their morphologies. Elliptical galaxies are smooth, featureless objects, appearing as ellipses in the images [3]. Spiral galaxies have disc-like structures and generally have 2 spiral arms. The third category from the Hubble galaxy classification scheme is irregular galaxies, which have very distorted shapes without the spiral arms or galactic bulge of spiral galaxies [4].



Fig. 1. Hubble galaxy classification scheme (Hubble 1926).

Deep learning has achieved significant results and resulted in a huge improvement in visual detection and recognition for cases with many categories [5]. Raw image data are used as input for deep learning without the need of expert knowledge for the optimization of the segmentation parameter or feature design. Features are not designed by human experts but are learned directly from the data via deep neural networks. Deep learning methods learn multiple levels of features by transforming the feature at one level into a more abstract feature at a higher level [6].

The idea of a convolutional neural network (CNN) is not recent. In 1998, CNN achieved promising results for handwritten digit recognition [7]. However, this system notably failed due to memory and hardware constraints as well as the absence of a large training dataset [8]. The system was unable to scale to much larger images. With the huge increases in the processing power and memory size and the availability of powerful GPUs and large datasets, it is now possible to train deeper, larger and more complex models. Machine learning researchers had been working on learning models that can learn and extract features from images.

The rest of the paper is organized as follows. Section (2) is an introduction to deep learning. Section (3) shows the related works. Section (4) discusses the proposed Deep Galaxies CNN architecture. Section (5) illustrates the data set and how to overcome the overfitting problem. The experimental results and environment are discussed in Section (6). Finally, Section (7) summarizes the main findings of this paper.

II. DEEP LEARNING

The main objective of deep learning algorithms is to detect many levels of distributed impersonations. These algorithms are a branch of machine learning algorithms. Recently, many deep learning algorithms have been used to solve classical artificial intelligence problems [9], with the main goal of learning high level abstractions from data. This learning is done by utilizing hierarchical architectures, which is an emerging approach and has been widely applied in traditional artificial intelligence domains, such as semantic parsing, transfer learning, natural language processing and computer vision [10]. The rise of deep learning today is derived from three substantial factors: the increase of chip-based processing capabilities, the lowered cost of computing hardware, and the huge advances in machine learning algorithms.

A. Neural Networks

Feed-forward neural network architectures with multiple layers are used by many deep learning architectures. The neurons in one layer are connected to all the neurons of the subsequent layer. All layers except the input and output layers are conveniently called hidden layers.

Artificial neurons are represented as mathematical equations in most artificial neural networks. These equations model the biological neural structure of the artificial neurons [11]. Let $x = (x_1, x_2, x_3, \dots, x_n)$ be a vector of the inputs for a given neuron, let $w = (w_1, w_2, w_3, \dots, w_n)$ be a vector of the weights, and let b be the bias. The output of the neuron is according to [12] will be

$$y = \sigma(w \cdot x + b) \quad (1)$$

where σ represents the activation function.

B. Nonlinearity as an Activation Function

The most popular activation function is the rectified linear unit (ReLU), the purpose of which is to introduce the nonlinearity of any operation, such as the convolution operations. ReLUs generally allow a faster training of deep neural networks with many hidden layers. According to [13], the use of the ReLU nonlinearity with their CNN has decreased the training time of their network significantly, making it six times faster than the equivalent CNNs with hyperbolic tangent nonlinearity. The ReLU is represented by equation (2). Figure 2 shows the graphical representation of ReLU.

$$\sigma(x) = \max(0, x) \quad (2)$$

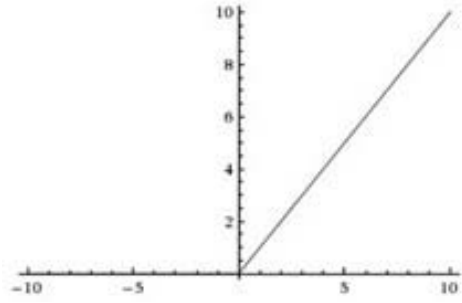


Fig. 2. The ReLU operation

C. Convolutional Neural Networks

The CNN is the most common type of deep, feed-forward neural network and is one of the most notable deep learning approaches. Recently, this type of neural network has become one of the popular tools in the computer vision community. A typical CNN has two types of layers as the main components of its first stages [14]. These layers are convolutional layers and pooling layers. In CNNs, multiple layers are trained in a robust manner.

The input of any convolutional layer is an image, and the output channels of each layer are called feature maps. Each feature map is convolved with a set of weights called filters and non-linearities, such as ReLU, which is applied to the weighted sum of these convolutions in order to produce a feature map. Different sets of filters are used by different feature maps [15]. The same set of filters are shared among all neurons in a feature map. Mathematically, a sum of the convolutions is used instead of the dot product in Equation 3. Thus, the k -th feature map is given by

$$y^k = \sigma(\sum_m w_m^k * x_m + b^k) \quad (3)$$

where we sum over the set of input feature maps, $*$ is the convolution operator, and w_m^k represent the filters.

The pooling layer reduces the spatial dimension of the representation given by the convolutional layer, which results in a decrease in the number of parameters and the number of computations within a network. Pooling works independently on every depth slice of its input, having a stride parameter similar to that of a filter of a convolutional layer. Pooling usually applies the MAX operation.

D. Gradient Descent (GD)

The learning process of neural network is based on searching for the combination of learnable parameters providing the lowest errors of the loss function. The gradient of the loss function is used to find the direction along with the weight vector [7]. This direction will be mathematically guaranteed to be the direction of the fastest descent of the loss function, as shown in equation (4)

$$w_i: w_i - \alpha \frac{\partial}{\partial w_i} L \quad (4)$$

where α represents the learning rate. The main functionality of this equation is for manually changing the gradient update [16]. During actual training, weights tend to increase far too much in each iteration. This increase would make them diverge and “over correct”.

The gradient descent is considered an optimization method that updates all the weights at once after running through all the samples in the training dataset once (this is called an epoch). However, its alternative, the stochastic gradient descent (SGD), updates the weights progressively after a subset of the training samples from the training dataset [16].

E. Back-propagation

The learning process of neural network is called back-propagation, which uses the gradient descent method to search for the minimum of a loss function [17]. The combination of weights with a minimum loss function is the solution of the learning problem. The back-propagation has two repeating phases. First, the network is given an input vector. This vector is propagated through the whole network resulting in an output. The network output is compared with the desired output by using a loss function, which calculates the error values for every neuron in the network, starting at the output layer and propagating through the whole network. The phase consists of updating all the weights by the desired optimization function [18].

III. RELATED WORKS

Prior researchers did not achieve satisfying results. In [19], the authors performed automated morphological galaxy classification. The classification was based on machine learning and image analysis. They depended on feed-forward neural networks and the local weighted regression method for classification. The achieved accuracy was approximately 91%.

In 2013, [20] used a Naïve Bayes classifier and random forest classifiers for morphological galaxy classification. The achieved accuracy was approximately 91% for the random forest classifiers and 79% for the Naïve Bayes classifier.

The authors in [21] proposed a method using a supervised machine learning system based on the nonnegative matrix factorization for images of galaxies in the Zsolt frei catalogue. The achieved accuracy was approximately 93%. In 2017, [22] proposed a new automated machine supervised learning astronomical classification scheme. The schema was based on the nonnegative matrix factorization algorithm. The accuracy of the new scheme was approximately 92%.

IV. PROPOSED NEURAL NETWORK ARCHITECTURE

The proposed architecture of galaxies morphology classification is introduced in detail in Figure 3 and Figure 4. Figure 3 illustrates the layers of the proposed architecture, while Figure 4 visualizes the proposed architecture via a graphical representation. The architecture consists of 8 layers, including one main convolutional layer for feature extraction, followed by two principle fully connected layers for classification. The first layer is the input layer. The second layer is the convolution layer. The third layer is a ReLU, which is used as the nonlinear activation function.

An intermediate pooling with subsampling is performed in the fourth layer. Then, layer five is a fully connected layer with 24 neurons, with a ReLU activation function. The last fully connected layer has 3 neurons and uses a soft-max layer to obtain the class memberships as illustrated in Figure 3.

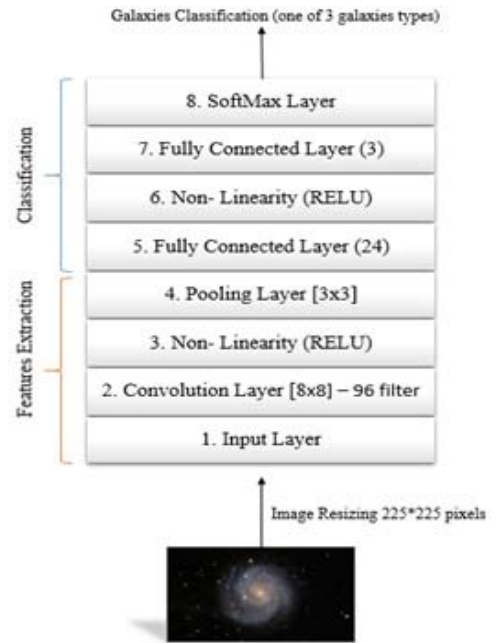


Fig. 3. Detailed layer descriptions for the proposed deep CNN architecture

V. GALAXIES DATASET

The dataset used was taken from the FIGI catalogue [23]. This catalogue dataset consists of more than 11,000 images and contains samples from different Hubble types galaxies. The catalogue combines data from standard surveys and catalogues (Sloan Digital Sky Survey, Value-Added Galaxy Catalogue, Principal Galaxy Catalogue, HyperLeda, and the NASA Extragalactic Database) [22].

The images used in the training and testing phase were selected according to the availabilities of the captured images.

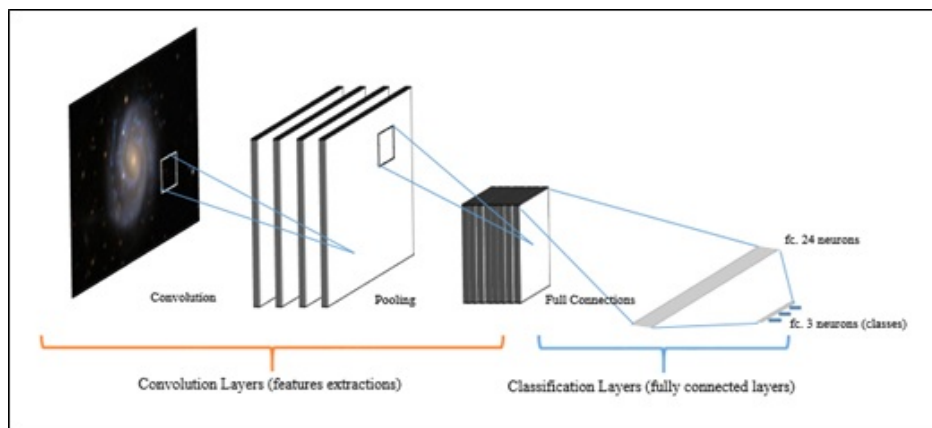





Fig. 4. Abstract view of the proposed deep CNN for galaxies classification

TABLE I. GALAXY TYPES WITH NUMBER OF TRAINING, VALIDATION AND TESTING IMAGES

Galaxy Type	Sample Image	Total Images	Training Set	Validation Set	Testing Set
Elliptical		1851	1203	278	370
Spiral		1739	1130	261	348
Irregular		648	421	97	129
Total		4238	2755	636	847

The sizes of the images vary in width and height. Table 1 illustrates the types of galaxies along with the number of training, validation and testing images for each galaxy type, including the augmented images.

A. Overfitting

Our proposed architecture has a huge number of learnable parameters compared to the number of images in the training set. Because of the vast difference between the learnable parameters and number of images in the training set, the model is very likely to overfit. In this section, the common techniques for overcoming overfitting are introduced.

B. Data Augmentation

The most common method for overcoming overfitting is to increase the number of images used in training by applying label-preserving transformations [13],[24]. Additionally, the data augmentation schemes are applied on the training set to make the resulting model more invariant to rotation, reflection and small noise in values of a pixel to be more robust other than

the previous work in [25]. To apply this augmentation, each image in the training data is transformed as follows:

Reflection: Each image is flipped horizontally with a probability of 0.3 to exploit mirror symmetry.

Rotation: Each image is rotated symmetrically and randomly by multiples of 45. However, the image rotation does not cause any changes of the galaxy type.

Zoom: Each image is magnified, as illustrated in Figure 5.

C. Visualizing the Architecture

Visualizing the feature extraction and classification layers in the proposed deep neural architecture will allow them to be better understood. Figure 6 shows the different images from applying the first convolution layer with 96 filters to the input image. In the classification layers, the first fully connected layer and its ReLU will produce galaxy images from the previous output images from the first convolution layer, as illustrated in Figure 7.

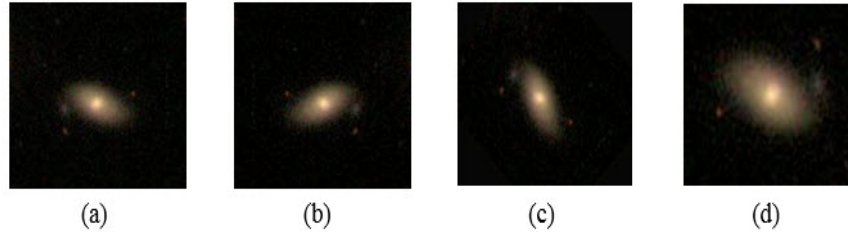


Fig. 5. (a) Raw image, (b) reflected image, (c) image rotated by 45 degrees (d) and the zoomed image

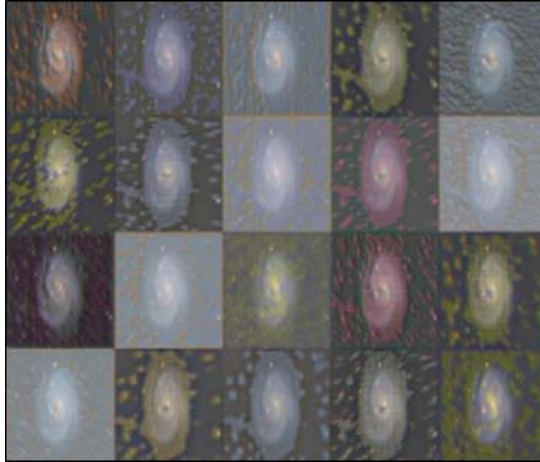


Fig. 6. Typical first convolutional and ReLU layer feature visualization.

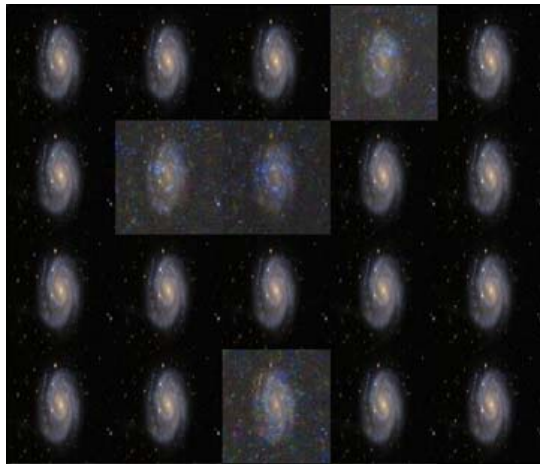


Fig. 7. Typical first fully connected layer visualization

VI. EXPERIMENT ENVIRONMENT AND RESULTS

The proposed architecture was developed using a software package (MATLAB). The implementation was CPU specific. All experiments were performed on a server with an Intel Xeon E5-2620 processor (2 GHz) and 96 GB of RAM.

To measure the accuracy of the proposed architecture for classifying galaxies types using deep convolutional neural networks, 5 different trials were performed, and the median accuracy was studied. Table 2 shows the comparative results of the proposed architecture and other related works. The proposed

architecture trained over 4238 images, including augmented images.

TABLE 2. COMPARATIVE RESULTS FOR GALAXY CLASSIFICATIONS

Related work	Year	Description	Accuracy
[19]	2004	Used neural network and a locally weighted regression for classification.	91%
[20]	2007	Used the Naïve Bayes classifier, rule-induction algorithm C4.5 and random forest.	91.64%
[21]	2016	Used a method based on nonnegative matrix factorization for images of galaxies from the Zsolt frei catalogue.	93%
[22]	2017	Used an automated machine supervised learning astronomical classification scheme based on the nonnegative matrix factorization algorithm.	92%
Proposed Architecture	2018	Used deep convolutional neural networks.	97.772%

VII. CONCLUSIONS

The classification of galaxies based morphologies is considered one of the motivating topics of interest to researchers over the past few years. The development of deep convolutional neural network science has also attracted more researchers into this field. In this paper, a robust deep convolutional neural network architecture for galaxies classification was introduced. This work is an extended version of “Deep Galaxy: Classification of Galaxies based on Deep Convolutional Neural Networks.” The Hubble classification was the main interest of research, as it allows galaxies to be classified based on their morphological features as one of three types: Elliptical, Spiral, and Irregular. The proposed architecture consisted of 8 layers, including one main convolutional layer for features extraction with 96 filters and two principle fully connected layers for classification. The architecture was then trained using 4238 images. Image augmentation techniques were applied to the training data and included rotation, reflection, cropping and Gaussian noise. These techniques decreased the overfitting problem of the proposed architecture. The testing accuracy was

97.772%. This accuracy is the median accuracy calculated from 5 different runs. The comparative results were introduced, and the accuracy achieved in the present work outperforms those of other related works.

VIII. REFERENCES

- [1] M. Abd Elfattah, N. El-Bendary, M. A. Abu Elsouid, A. E. Hassanien, and M. F. Tolba, "An intelligent approach for galaxies images classification," in 13th International Conference on Hybrid Intelligent Systems (HIS 2013), 2013, pp. 167–172.
- [2] M. Abd Elfattah, N. Elbendary, H. K. Elminir, M. A. Abu El-Soud, and A. E. Hassanien, "Galaxies image classification using empirical mode decomposition and machine learning techniques," in 2014 International Conference on Engineering and Technology (ICET), 2014, pp. 1–5.
- [3] A. Hocking, Y. Sun, J. E. Geach, and N. Davey, "Mining Hubble Space Telescope images," in 2017 International Joint Conference on Neural Networks (IJCNN), 2017, pp. 4179–4186.
- [4] M. D'Onofrio et al., "The Anatomy of Galaxies," Springer, Cham, 2016, pp. 243–379.
- [5] N. Lu, T. Li, X. Ren, and H. Miao, "A Deep Learning Scheme for Motor Imagery Classification based on Restricted Boltzmann Machines," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 25, no. 6, pp. 566–576, Jun. 2017.
- [6] P.-H. Liu, S.-F. Su, M.-C. Chen, and C.-C. Hsiao, "Deep learning and its application to general image classification," in 2015 International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS), 2015, pp. 7–10.
- [7] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.
- [8] H. Habibi Aghdam and E. Jahani Heravi, Guide to Convolutional Neural Networks. Cham: Springer International Publishing, 2017.
- [9] I. Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, and C. McCool, "DeepFruits: A Fruit Detection System Using Deep Neural Networks," Sensors, vol. 16, no. 8, p. 1222, Aug. 2016.
- [10] S. Bargoti and J. Underwood, "Deep fruit detection in orchards," in 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017, pp. 3626–3633.
- [11] C. C. Aggarwal, Data classification : algorithms and applications. CRC Press, 2014.
- [12] Z. Le and S. P.N., "A survey of randomized algorithms for training neural networks," Information Sciences, vol. 364–365, pp. 146–155, Oct. 2016.
- [13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Proceedings of the 25th International Conference on Neural Information Processing Systems. Curran Associates Inc., pp. 1097–1105, 2012.
- [14] H. Jiang and E. Learned-Miller, "Face Detection with the Faster R-CNN," in 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), 2017, pp. 650–657.
- [15] Q. Liu et al., "A Review of Image Recognition with Deep Convolutional Neural Network," in International Conference on Intelligent Computing, Springer, Cham, 2017, pp. 69–80.
- [16] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in Neural networks: Tricks of the trade, Springer, 2012, pp. 437–478.
- [17] Y. Wu, Y. Fang, X. Ren, and H. Lu, "Back propagation neural networks based hysteresis modeling and compensation for a piezoelectric scanner," in 2016 IEEE International Conference on Manipulation, Manufacturing and Measurement on the Nanoscale (3M-NANO), 2016, pp. 119–124.
- [18] R. Fontanella, D. Accardo, E. Caricati, S. Cimmino, and D. De Simone, "An extensive analysis for the use of back propagation neural networks to perform the calibration of MEMS gyro bias thermal drift," in 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS), 2016, pp. 672–680.
- [19] J. De La Calleja and O. Fuentes, "Machine learning and image analysis for morphological galaxy classification," Monthly Notices of the Royal Astronomical Society, vol. 349, no. 1, pp. 87–93, Mar. 2004.
- [20] M. Marin, L. E. Sucar, J. A. Gonzalez, and R. Diaz, "A Hierarchical Model for Morphological Galaxy Classification," in Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference, 2013, pp. 438–443.
- [21] I. M. Selim, A. E., and B. M.El, "Galaxy Image Classification using Non-Negative Matrix Factorization," International Journal of Computer Applications, vol. 137, no. 5, pp. 4–8, Mar. 2016.
- [22] I. M. Selim and M. Abd El Aziz, "Automated morphological classification of galaxies based on projection gradient nonnegative matrix factorization algorithm," Experimental Astronomy, vol. 43, no. 2, pp. 131–144, Apr. 2017.
- [23] A. Baillard et al., "The FIGI catalogue of 4458 nearby galaxies with detailed morphology," Astronomy & Astrophysics, Volume 532, id.A74, 27 pp., vol. 532, Mar. 2011.
- [24] Schutter A. and Shamir L., "Galaxy morphology — An unsupervised machine learning approach," Astronomy and Computing, vol. 12, pp. 60–66, Sep. 2015.
- [25] Khalifa, N. E. M., Taha, M. H. N., Hassanien, A. E., & Selim, I. M. (2017). Deep Galaxy: Classification of Galaxies based on Deep Convolutional Neural Networks. arXiv preprint arXiv:1709.02245, Sep. 2017.