Politechnika Krakowska
im. Tadeusza Kościuszki

Wydział Fizyki, Matematyki i Informatyki

Eryk Kozłecki

Numer albumu: 104871

Morfologiczna klasyfikacja galaktyk przy użyciu algorytmów głębokiego uczenia

Morphological Classification of Galaxies Using Deep Learning Algorithms

Praca magisterska
na kierunku Fizyka techniczna

Promotor:
Dr. Radosław Kycia

Uzgodniona ocena:.....

.....
podpisy promotora i recenzenta

Kraków, 2017

Master Thesis

Morphological Classification of Galaxies Using Deep Learning Algorithms

Author: Eryk Kozłecki
Supervisor: dr Radosław Kycia

Tadeusz Kościuszko University of Technology
Kraków, Poland

June, 2017

Abstract

The main goal of this thesis was to find, examine and describe the most suitable way of using deep learning methods that will be able to help (or completely automate) the process of recognizing and classifying several types of galaxies based on photos drawn from the Sloan Digital Sky Survey.

Morphological analysis of photos of galaxies is very useful for studying galaxy formation and evolution, but the amount of data that Sloan Digital Sky Survey and other surveys have provided us is way too superior - manual classification is a tedious task for researchers and volunteers, so there is a great need in finding proper algorithms that may help astronomers at this venture.

Due to vast growth in computational power and significant increase in interest in machine learning models in almost every area of life, the next logical step is to use this potential and try to create autonomous classification methods that will help us in better understanding the world we live in.

Keywords: Machine Learning, Deep Learning, Deep Neural Network, Galaxy Morphological Classification, Sloan Digital Sky Survey.

Acknowledgments

This thesis would not have been possible without the support of a number of many individuals - my thanks and appreciation to all of them for making this thesis possible. I would like to express my sincere gratitude to my supervisors dr Radosław Kycia. Without his enthusiasm, guidance, invaluably constructive criticism and support this thesis would hardly have been completed.

Contents

List of Figures

1

Introduction

Structural characteristics of galaxies has been a long-term goal in cosmology - it is an important area of interest in the large-scale study of the observable universe. Galaxy classification, especially morphological classification is the first and maybe the most important of the steps towards understanding of the origins and the evolution processes of galaxies, and the evolution of the Universe in general. Galaxy classifications are important for two major reasons [35]:

- Provide comprehensive catalogues for astronomical and statistical studies;
- Study corelations between structures of the galaxies and processes occurring during the early stages of the Universe.

To make such studies possoble researchers from all around the world gathered much information on the content and character of the universe structure through sky surveys and mappings of the various wavelength bands of electromagnetic radiation. One of the most important projects in this field was The Sloan Digital Sky Survey (SDSS) - the observation program was launched in 2000 and has now led us to the discovery of almost 930 000 galaxies, by covering over 35% of the sky [22].

In the recent years astronomy has become an immensely data-intensive field. It is not hard to guess that such a huge amount of data requires a completely new approach that can be helpful in analysis these datasets. The

amount of images of galaxies produced every year is impossible to be reviewed by humans - this problem creates a need for techniques that could automate the difficult problem of classification. Machine learning methods seem to be most suitable for this kind of work.

Machine learning techniques have been used in astronomy and astrophysics for more than twenty years, but only now, thanks to current state of technological advancement, we have access to larger training sets and almost unlimited computing resources, which can greatly improve accuracy and complexity of trained models. The most promising and suitable methods than can find use in cosmological researches are deep learning algorithms, especially those that are based on convolutional neural networks (CNN) [41].

The Sloan Digital Sky Survey led us to The Galaxy Zoo Project, a citizen, web-based science project that aimed to obtain morphological classifications for roughly a million objects, including galaxies. Creators of this project harnessed the power of the internet by recruiting members of the public and ask them to perform classifications by eye [11]. However, as data sets grow to contain billions of galaxies, approach of this kind becomes less and less feasible. In cases like that, the deep learning algorithms find their applications.

This diploma thesis is focused mostly on the algorithmic and computional part of the morphological classification of galaxies and deep learning methods in general. The main effort is development of efficient neural network-based algorithms capable of reliable object type classification based on data delivered by The Galaxy Zoo Project.

This thesis is organized as follows: The first parts contain general introduction to galaxy classification and description of machine learning systems. The next sections holds overall specification of used software technologies and datasets. Next we move on to the more accurate description of Artificial Neural Networks and after that we will discuss applied approach to the stated problem.

2

Galaxy morphological classification

Galaxies are gravitationally bound celestial systems composed, as current cosmology suggest, of billions of stars, stellar remnants, interstellar mediums and dark matter. Many of them are thought to have black holes at their centers. The Milky Way's central black hole, Sagittarius A, has a mass over four million times greater than the mass of the Sun [2][23].

Most of the stars are associated in a disks that are about 100 000 light years across in diameter and 3 000 light years thick and the most recent number of galaxies in the observable universe is estimated from 200 billion to 2 trillion or more, all of them contais more stars than all the grains of sand on Earth. Most of the galaxies are 1 000 to 100 000 parsecs (*"a parsec is defined as the distance at which 1 Astronomical Unit subtends an angle of 1 second of arc [1/3600 of a degree] - 1 parsec* $\approx$ *3.26 light years*" [6]) in diameter and are separated by distances on the order of megaparsec (millions of parsecs). This space between them is filled with a gas having an average density of less than one atom per cubic meter and the most of them are organized into groups, clusters, and superclusters. At the largest scale, these groups are generally arranged into sheets and filaments surrounded by immense spaces. The largest association of galaxies yet recognised is a cluster of superclusters, named Laniakea [30][35][38].

Galaxies form over billions of years, and can be marked according to their morphology – their shape and general visual appearance – which gives researcher much information about their evolution and composition. Mor-

phology is a reasonable starting point for understanding galaxies. Classifying galaxies into their morphological categories is very similar to classifying stars into spectral types and can carry on to important astrophysical insights. Galaxy morphological classification is strongly correlated with star formation history - galaxies where stars formation ceased billion years ago tend to look different from those where formation of the stars continue to the present time.

There are a few systems in use by which galaxies can be identified and classified by their morphologies, but the most famous is the Hubble sequence, contrived by Edwin Hubble and later expanded by Allan Sandage and Gérard de Vaucouleurs [35].
These scheme will be described in subsequent subsection.

2.1 Hubble sequence

The most commonly used classification system, both in professional astronomical studies as well as in amateur astronomy, is the model devised by Sir Edwin Hubble in 1936 an expanded by Allan Sandage and Gérard de Vaucouleurs in later years.

Hubble divided regular galaxies into four main classes, based on visual appearance of galaxy images stored on photographic plates:

- **ellipticals**: E0, E3 , E5, E7,
- **spirals**: S0, Sa, Sb, Sc,
- **barred spiral**: SBa, SBb, SBc,
- **lenticulars**: S0
- **irregulars**: Im, IBm, Irr I, Irr II.

This Hubble sequence schema is commonly referred to as the "Hubble Tuning Fork" and is traditionally illustrated as shown in the figure below:

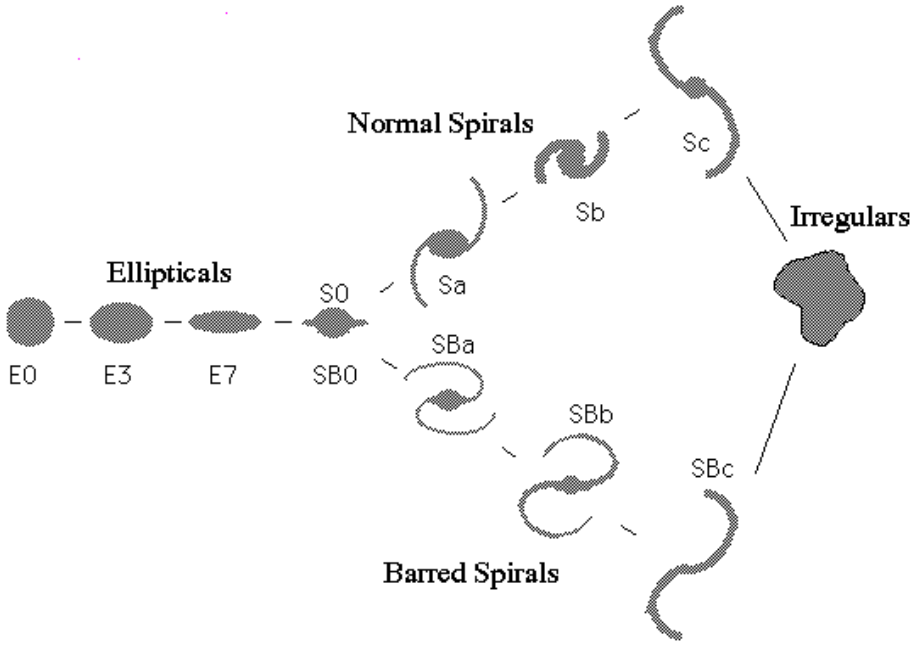

Figure 2.1: Hubble sequence schema [46]

.

Basically, the Hubble schema represents the presented rule by asking the following questions [46]:

1. Is there overall regularity or symmetry to the galaxy?
2. Is the light concentrated in the center of the galaxy?
3. Is there a disk or even disk's seed in the galaxy representation?
4. Are there any spiral arms in the galaxy image?

2.2 Classes of galaxies

Elliptical and lenticular galaxies are well known as "early-type" galaxies, and spirals and irregular ones are related to "late types".

It was thought that the disks of spiral galaxies were observed to be home of many young stars and areas of active star formation processes, while elliptical galaxies were composed of most old stellar populations. Current predictions suggests quite the opposite [40]: the early Universe seems to be dominated by spiral and irregular types. Currently favored idea of galaxy formation suggest that present ellipticals galaxies formed as a result of combination (link) between these early-stage type building blocks [40]. Barred spiral galaxies may also evolved from spiral galaxies, whose gas has been run-down, leaving no fuel for future star formation [29].

2.2.1 Ellipticals

Elliptical galaxies are mellow, amorphous systems with a continuously, slowly declining brightness distribution (progressing from the center), and with the lack of inflections, breaks, zones or structures and no sign of a disk. They appear elliptical in shape in photographic images and nearly all of them have the same color - look much the same at different wavelengths, because they are dominated by old stars and are also devoid of gas or dust.

They are indicated by the letter *E*, followed by an integer digit n, that represents their degree of ellipticity in the sky: $\epsilon = 1 - \frac{b}{a}$, where a and b are semimajor and semiminor axes of the ellipse:

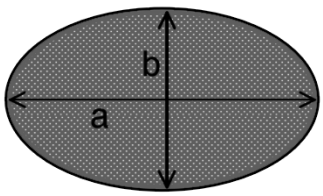

Figure 2.2: Ellipticity ancillary figure.

By convention, n is multiplied ten times bye the ellipticity ϵ of the galaxy and rounded to the nearest integer. Rounded ellipticals are classed as E0 and highly flattened ellipticals are classed as E7 [29][35].

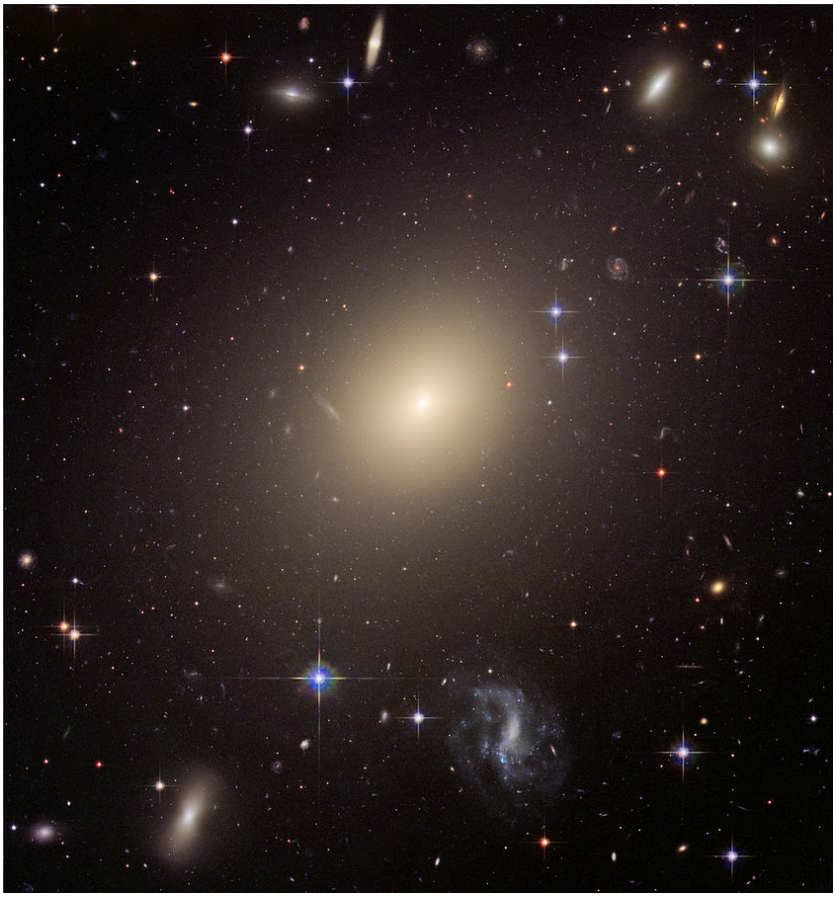

Figure 2.3: The giant elliptical galaxy ESO 325-G004 [29].

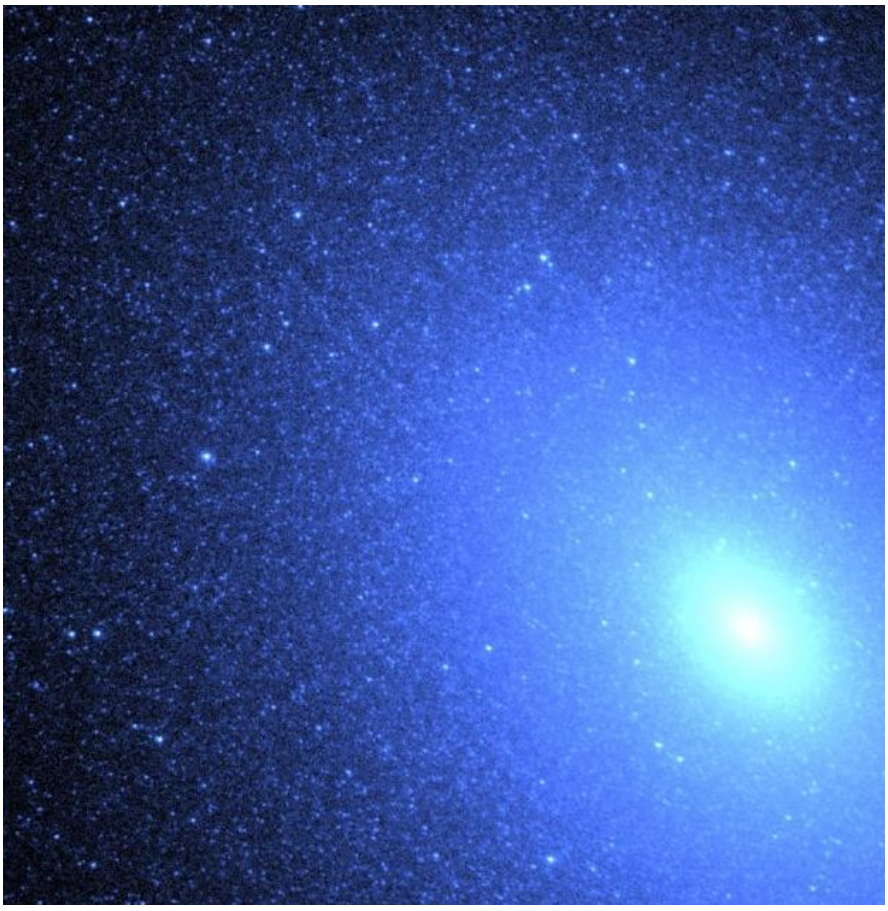

Figure 2.4: The dwarf elliptical galaxy M32 [46].

2.2.2 Spirals

A spiral galaxy contains a flattened disk, a central-concentrated aggregation of stars known as the bulge and a two-armed stars forming spiral structure. Almost half of all spiral types are observed with a bar-like structure, that is extending from the central, where the spiral arms begin [46].

The Hubble diagram contains two branches of the spiral types:

- the upper branch: **regular spirals galaxies** (**S**);
- the lower branch: **barred spirals galaxies** (**SB**).

Both types are further subdivided according to the detailed appearance in their internal structures:

- Sa (SBa) - smooth, tightly shaped spiral arms; bright and large central bulge;
- Sb (SBb) - less tightly shaped arms; less fainter bulge;
- Sc (SBc) - loosely shaped arms, consisting mainly of individual stellar clusters and nebulae; smaller and fainter bulge;
- Sd (SBd) - very loosely shaped, fragmentary arms; most of the luminosity is located in the arms (not in the the bulge).

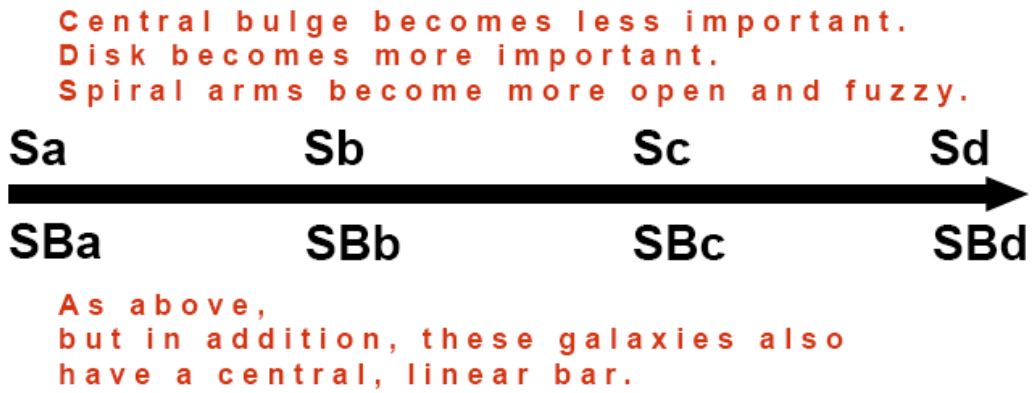

Figure 2.5: Two sequences of spiral galaxies [46].

Figure 2.6: The Pinwheel Galaxy (Messier 101/NGC 5457): a spiral galaxy classified as type Scd [29].

Figure 2.7: The barred spiral galaxy NGC 1300: a type SBbc [29].

2.2.3 Lenticulars

Lenticular galaxies (designated as S0) - transition class between ellipticals and spirals galaxies. S0 types are similar in appearance to an elliptical galaxy, consist as well a bright central bulge surrounded by an extended, disk-like structure (but have no visible spiral structure). The central component is, in most cases, the dominant source of light.

The lenticular type is difficult to distinguish from E0 elliptical type, making the classification of many galaxies highly uncertain. Lenticulars can also have a central bar (like the spiral ones), in which case they are labeled by SB0 [29].

Figure 2.8: The Spindle Galaxy (NGC 5866), a lenticular galaxy with a prominent dust lane in the constellation of Draco [29].

Figure 2.9: Spiral galaxy UGC 12591, classified as an S0/Sa galaxy [29].

2.2.4 Irregulars

The most of representatives of this class do not fit into the Hubble sequence, due to no regular structures: grainy, highly irregular agglomeration of luminous areas. They does not have noticeable symmetry and obvious central part.

Hubble defined a few classes of irregular type galaxies:

- **Irr I** -lack of a central bulge and spiral structure and having an asymmetric profile - they contain a lot individual clusters of young stars;
- **Irr I** - smoother, asymmetric look and are not clearly resolved into individual stars or clusters of young stars;
- **Im** - highly irregular galaxies.

Figure 2.10: The Large Magellanic Cloud (LMC) - a dwarf irregular galaxy. A satellite galaxy of the Milky Way [29].

Figure 2.11: Messier 82 (M82) - a highly irregular galaxy [29].

Irregular types are similar to spirals in having both old and young stars, as well as dust and ionized gas, but they are lacking in the spiral structure that triggers star formation process.

2.3 Visual summary

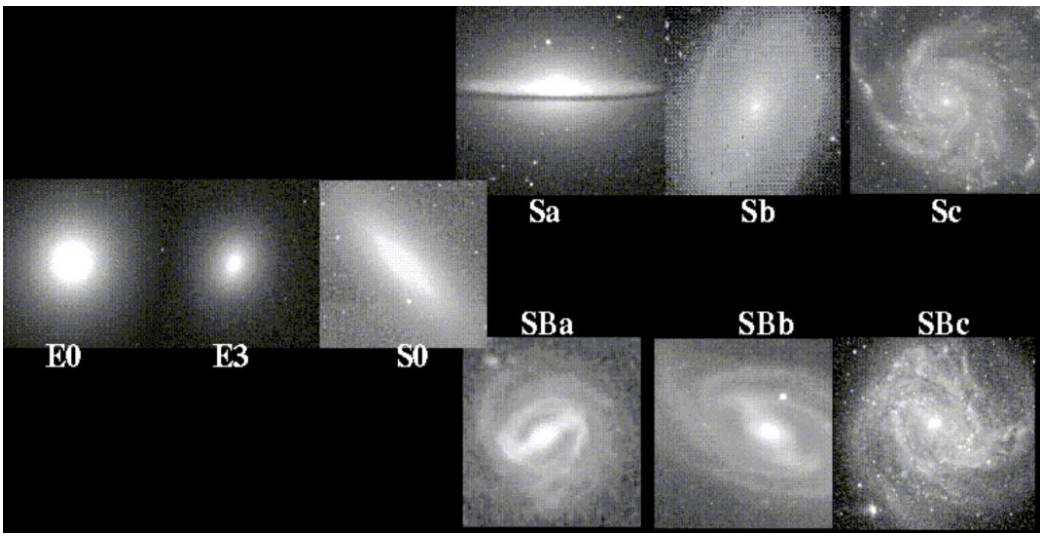

Figure 2.12: A visual summary of the Hubble model [18].

3

Artificial Intelligence, Machine Learning and Deep Learning

When programmable computers were first imagined, people wondered if they might ever become intelligent in the way we perceive this process. da Lovelace, an English mathematician and the doughter of Lord Bayron, considered the concept of "thinking machine" (1842), over a hundred years before a cumputer was built [1].

Todays artificial intelligence models, such as machine learning or deep learning, are well developed field with many practical applications, active research topics and extraordinary future goals. Intelligent software is about to understand speech and images, make diagnoses in medicine, automate routine labor and support scientific research [37].

Artificial Intelligence (AI), Machine Learning (ML) , and Deep Learning (DL) - these terms are related and overlapping with each other.
AI is rather general definition - it involves machines and algotirhms that can perform tasks that are characteristic to human intelligence, which includes things like problem solving, recognizing objects and sounds, understanding language, planning and learning.
While AI is the core, machine learning methods are simply a way of achieving Artificial Intelligence: ML is one subfield of AI. The core principle of these methods is that machines take data and use them to "learn" for themselves - at its most basic level, it is the method of using algorithms to analyze data,

learn from it, and then make a prediction and draw conclusions about the problem that we was trying to solve [42].
Deep Learning (or sometimes Deep Neural Nets) is one of many ways of achieving machine learning that was inspired by the functions and structure of the brain. DL uses Artificial Neural Networks (ANNs), algorithms that are trying to mimic and evolve the biological structure of the brain - interconnections of many neurons [34].

We can think of this differences like as a set of Matryoshka dolls that are nested within each other, beginning with the largest and working out. Artificial Intelligence is the main, biggest set, while Machine Learning and Deep Learning are subsets, grouped in each other:

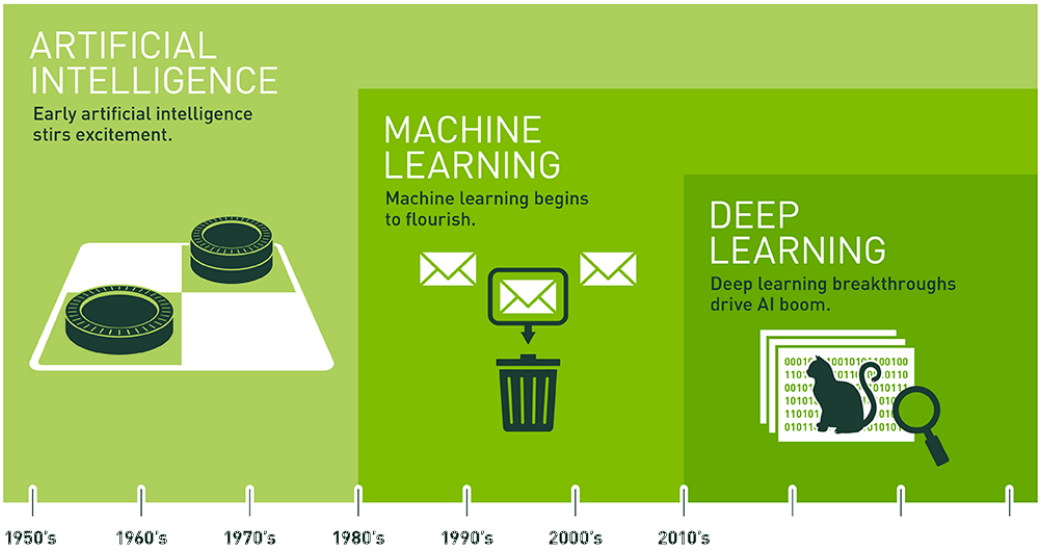

Figure 3.1: Relationship between AI, ML and DL [36].

3.1 Artificial Intelligence

"The science and engineering of making intelligent machines, especially intelligent computer programs."

- John McCarthy, *father of AI*

Artificial Intelligence is the widest way of thinking about advanced, computer intelligence.
In 1956 at the Dartmouth Artificial Intelligence Conference, the AI was described as: *"Every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it."* [44].

AI reffers to the ability of a computer program or a computer-enabled robotic system to proces information and develop outcomes in a manner simmilar to the thought processes of human beeings in learning, decision making and solving problems. The main goal of AI system is to develop structure capable of tracking complex problems in a ways simmilar to human logic and reasoning.

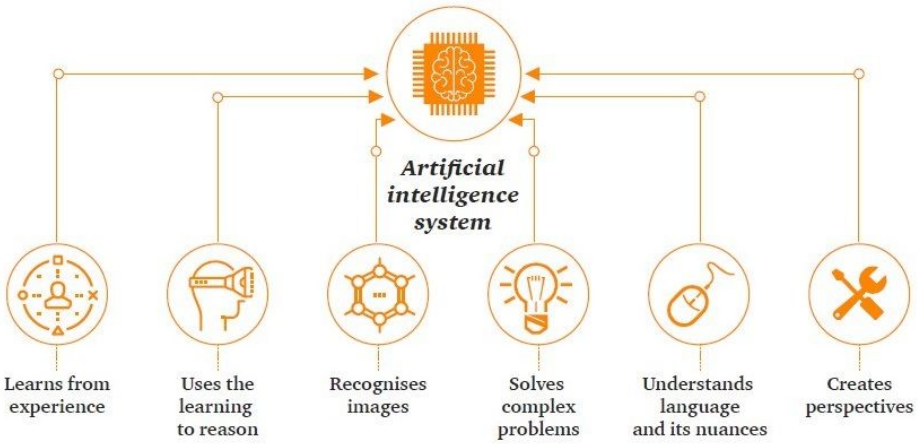

Figure 3.2: AI system [34].

The technology can be categorized into three major groups: [42]

- **general AI** - can all of the characteristics of human intelligence;
- **narrow AI** - can obtain some aspects of human intelligence: it is skilled at one specific task but lacking in other areas;
- **superintelligent AI** - system that surpasses humans in every field of knowledge.

AI has drastically evolved over the past few decades, to the point when we all use some part of it, even without knowing about the fact.

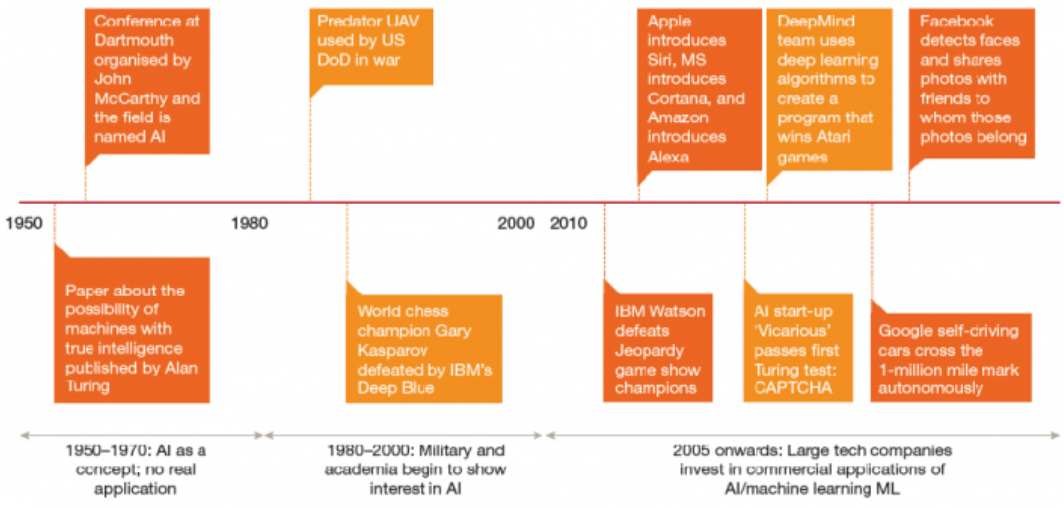

Figure 3.3: Evolution of AI. [34]

3.2 Machine Learning

Machine Learning is one of subfields of AI. It is a type of AI that simplify computer ability to learn from data and essentially learns by teaching itself to evolve as becomes exposed to new and changing inputs.

ML systems can quickly adopt trained knowledge gathered from large data sets and apply it on fileds like: translation, facial recognition, speech recognition, object recognition etc. ML allows systems to learn by recognizing patterns on its own and make predictions, unlike traditional, hand-coding software programs that requires specific instructions to complete a task.

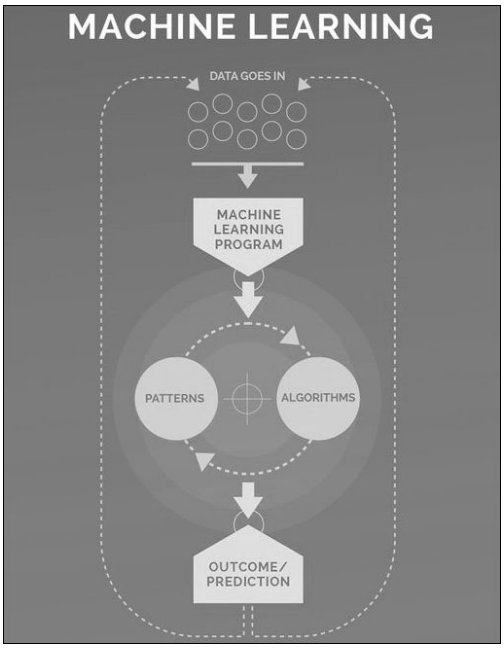

Figure 3.4: How ML works [34].

The main elements of machine learning algorithms are statistical and predictive analysis used to find patterns and hidden insights based on observed data from previous iterations, without being previously programmed on where to search for these patterns [34].

There ara few achine learning classification system but the most common is the one that divide ML into three groups depending on the nature of the learning inputs or outputs from a learning system, so it depends of types of problems and tasks.

Three main categories of ML [37]:

- **spervised learning:** - a computer is presented with example inputs and outputs data;
- **unsupervised learning** - no example result are given to the learning algorithm;
- **reinforcement learning** - a program interacts with a dynamic environment and learn on its own mistakes.

Deep learning is only one of many ways of achieving machine learning. The other popular methods are [31]:

- decision tree learning,
- association rule learning,
- artificial neural networks,
- inductive logic programming,
- support vector machines,
- learning classifier systems.
- bayesian networks,
- representation learning,
- reinforcement learning,
- similarity and metric learning,
- sparse dictionary learning
- genetic algorithms,
- clustering,
- rule-based machine learning,
- learning classifier systems.

3.3 Deep Learning

Deep learning is one of many approaches of machine learning and at the same time it is a new area of ML research, and thanks to that we are getting closer to general artificial intelligence.

It relates to study of deep neural networks in the human brain, namely the interconnecting of many neurons - the deep learning tries to emulate the functions of inner layers of the human brain by creating knowledge from multiple layers of information processing. Thanks to this approach, the more data is added, the capabilities of system gets better. According to that, Artificial Neural Networks (ANNs) are algorithms that try to mimic the biological structure of the brain, but, unlike the biological brain where a neuron can only connect to any other neuron within a specified physical

distance, artificial neural networks have discrete layers between neurons and can choose directions of data propagation.

Each layer can choose a specific feature to learn (for example curves or edges in image recognition). It is this layering process that gives deep learning its name - the depth is created by using multiple layers [36].

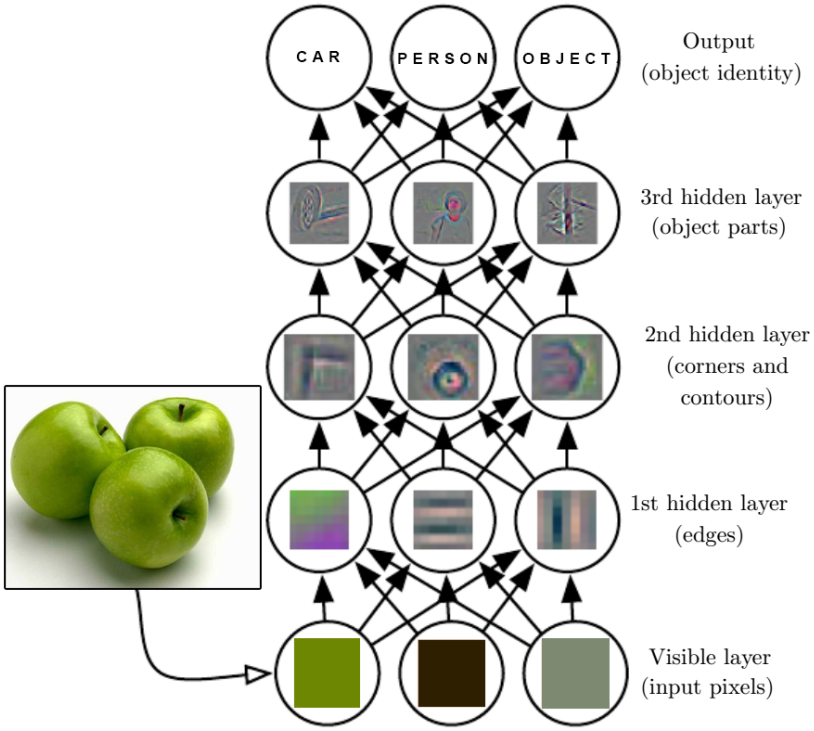

Figure 3.5: Illustration of a deep learning model [37].

Deep Learning, in the most basic terms, can be explained as a probability system. Based on a dataset, it is able to make decisions and predictions with a some degree of certainty by assigning a weighting to input of each artificial neurons. Each weight is correlated to probability of how correct or incorrect was the prediction. The final result is determined by the total of those probabilities [36].

3.4 Applications

One of the most useful applications of Machine Learning is image recognition, which in most cases performs better than humans: from the range of identifying simple objects like traffic signs or handwritings to recognizing tumors in MRI scans and indicators for cancer in blood cells. Many people may do not know that they encounter machine learning applications in their everyday lives. Algorithms in social media services are used to trend important topics or hashtags. E-commerce corporations build models that predict customers behavior, e-mail providers builds anty-spam filters usig ML [36]. By te use of AI methods, scientists build models to predicts strokes and seizures, identifies heart failure, predict hospital readmissions, helps synthesis of new compounds and much more [9].

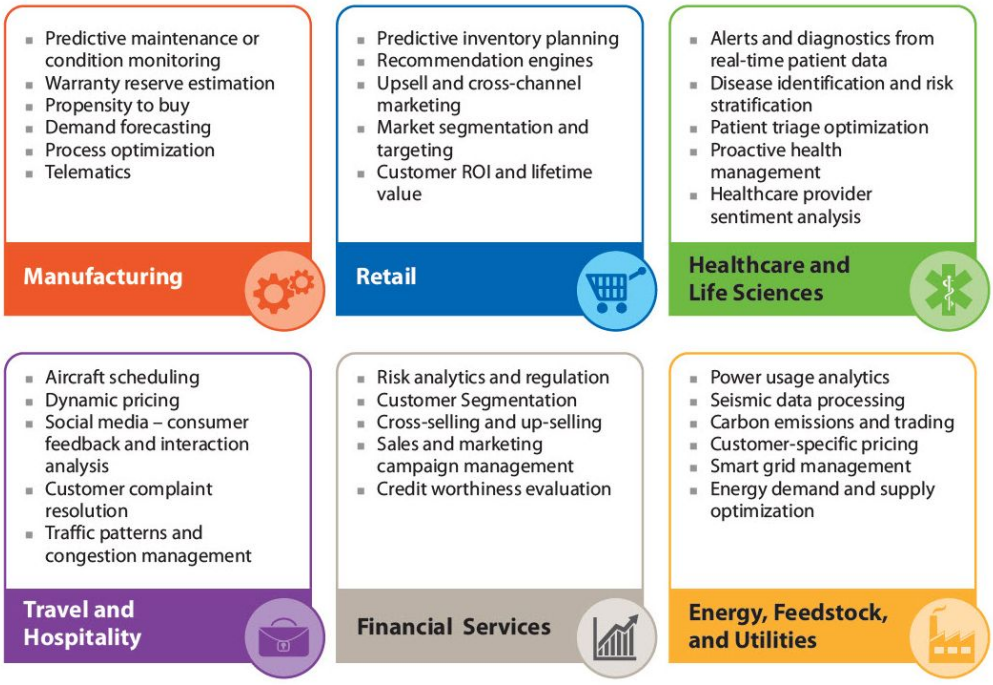

Figure 3.6: Illustration of a deep learning model [9].

4

Applied technologies

In this section the computer tools that will be useful in constructing our deep learing system will be described. We will start from Python and NumPy and then we will move to description of pure machine learning libraries.

4.1 Python programming language

Python is a high-level programming language that is widely used for general-purpose programming, created and released in 1991 by Guido van Rossum. Due to its highly readable syntax, open source philosophy, support for multi-paradigm programming and use of dynamic typing it has become one of the most commonly used programming language by the scientific community. However, its main advantage seems to be the support for variety of libraries, suplements and frameworks created for deep learning research [20].

4.1.1 NumPy

NumPy is the fundamental library for scientific computing with Python language [17].

Among other things, it provides:

- N-dimensional array objects manimulation tools,
- powerful linear algebra, tensors manipulation and random numbers ca-

pabilities,

- multi-dimensional containers of generic data,
- possibilities to define arbitrary data-types.

4.1.2 TensorFlow

TensorFlow is an open source Python and C++ library for machine learning across a range of tasks. It was developed in 2015 by Google Brain Team mainly for training neural networks to detect and solve patterns and correlations, can run on machines with multiple CPUs and GPUs and it available on all popular operating systems. The library is still being developed, optimized and improved [24].

4.1.3 Keras

Keras is a powerful, open source and easy-to-use Python library than can run on the top of TensorFlow and is suitable for developing and evaluating deep learning models. It includes efficient numerical subsystem that allows us to define and train neural network models. The library contains as well variety of implementations for commonly used neural network blocks (such as layers, objectives etc.) and host tools for image and text processing data. Keras was originally developed in 2015 by François Chollet [14].

4.2 CUDA technology

Artificial Neural Networks architectures have been developing for more than a dozen years, but only now we can notice drastic jump in their performance and capabilities. The reson for this is progress that has been made in parallel computing techniques, especially in GPU-powered calculations.

CUDA (Compute Unified Device Architecture) is a parallel computing platform and application programming interface system that allows software developers to use graphics processing units (GPUs) for general purpose processing calculations. It was developed in 2007 by Nvidia corporation, the largest manufacturer of graphics chips and since then the system is widely supported across all parrarel programing cloud-based virtual machines [28].

5

Datasets

5.1 Sloan Digital Sky Survey

Sloan Digital Sky Survey (SDSS) is the major and most ambitious astronomical survey that has been ever undertaken. It provides a multi-filter imaging and spectroscopic redshift map of about a million galaxies and quasars, that covers over 35% of the sky [22].

The survey was performed by using a 2.5-m wide-angle optical telescope, which was located at Apache Point Observatory in New Mexico, with the cooperation of more than 40 institutions from all over the world. The camera was retired in 2009 and since then the telescope has worked entirely in spectroscopic mode. The project was named in honor the Alfred P. Sloan Foundation - the contributor of substantial funding.

The single shot covers about 1.5 square degree of the sky and is recorded in five colors by a CCD camera. Data collection began in 1998 and upon this day brought the discovery of 930 000 galaxies and over 120 000 quasars [32].

The survey is conducted in stages:

- SDSS-I - 2000–2005,
- SDSS-II - 2005–2008,
- SDSS-III - 2008–2014,
- SDSS-IV - 2014–2020.

The SDSS-IV survey will also include observations from the southern hemisphere by the Irénée du Pont Telescope from Las Campanas Observatory, Chile.

Sloan Digital Sky Survey releases the available data over the Internet, mainly by the **SkyServer** [21].

5.2 Galaxy Zoo

Galaxy Zoo [11] is a worldwide, crowdsourced astronomy project (a citizen science project) which involve people to help scientists in the task of morphological classification of large numbers of galaxies obtained from the Sloan Digital Sky Survey to determine the different aspects and separate galxies into classifications [39].

The Galaxy Zoo project has gone through few stages:

1. The first stage focused on determining if a galaxy was elliptical, spiral or a merger of two galaxies.

2. Galaxy Zoo 2 asked volunteers for more details, include measurements of the bulge size, structure of spiral arms or presence of bars.

3. The present Galaxy Zoo challenges combines the newest imaging from the SDSS with the Hubble's CANDELS project[4] that is able to take ultra-deep images of the Universe.

5.2.1 Galaxy Zoo 2

Morphological data for stage two of Galaxy Zoo were collected by a web-based interface. [11] A Volunteer needed to register with a username and the interface guided him through whole classifications process by using a nested decision tree, consists of 11 questions with 2-7 responses to them.

Each user's classification is the outcome of a particular path down a decision tree. Multiple volunteers (about 50) classified the same galaxy, generating multiple paths through the decision tree and and assigned probabilities for branch (node).
Based on these probabilities researchers were able to produce final outcome generated for each galaxy image [13].

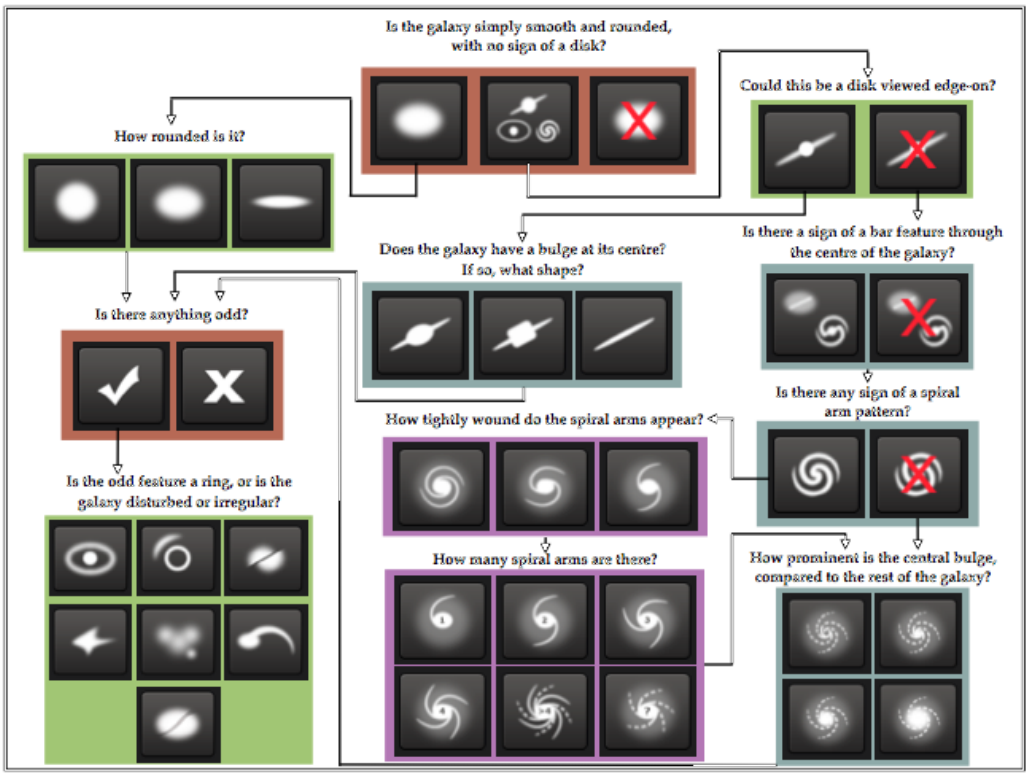

Figure 5.1: Decision tree used for the Galaxy Zoo 2 [39].

Throuh the 14-months process, 83 943 volunteers obtained 16 340 298 classifications of 304 122 galaxies [39].
Data gathered from this phase of classification is being used for the purpose of this thesis.

6

Deep learning and Artificial Neural Networks

Deep learning methods are based on a class of algorithms called Artificial Neural Networks (ANNs) constructed of many (deep) layers. Such networks existed for many decades, but only now, due to Geoffrey Hinton's breakthrough researches from the border of cognitive psycholoy and computer science [12][33], and increase in computing capabilities using GPUs procesor units, we are able to train deep architectures with such accuracy, that allows us to use the models in computer vision, natural language processing or automatic speech recognition [43].

The idea of deep learning is to create a model that tries to express given data at multiple levels of abstraction and by that, automatically find accurate representations from the input data itself. Such models consist of several hierarchy-based layers. Each layer have a gradually more abstract image of the input data than the previous layer - the process is done by calculation a nonlinear transformation of inpus. Parameters of the transformation are computed by training models on a specific dataset [45].

6.1 Artificial Neural Networks

Artificial neural networks (ANNs), as its name suggests, are a family of machine learning techniques inspired by biological neural conecctions - modeled

and mapped after the brain structures. ANNs contain of a set of deep learning units called neurons (named after biological neurons) that learn how to convert input signals into compatible, corresponding outputs [5].

Biological Neural Networks consist of Biological Neurons - the core computational unit of the human brain structure. A single biological neuron contains: a cell body, an axon and dendrites and by these components such neuron can processes and transmit an information to other connected neurons by emitting electrical signals. Each neuron can produce output signals alongside its axon and inputs from its dendrites - Different axons and dendrites are connected via synapses and these connections shape a human brain by the form of biological network.
The human nervous system consist of approximately 86 billion neurons that are connected with approximately 10^{15} synapses [7].

Artificial Neural Networks are inspired by its biological equivalent - they try to explain and formulate the biological model in a computational form. In the basic model, an artificial neuron can hold a finite number of inputs (with weights assigned to them) and a transfer (activation) function that sends the information spikes alongside the axon: all the informations get summed in the cell body. If the sum exceeds a established threshold, the neuron sends the signal and whole operation repeats. The output data of a single neuron is the outcome of the transfer function applied to the weight sum of inputs. Artificial neurons are combined with each others and together they form artificial neural networks [7][5].

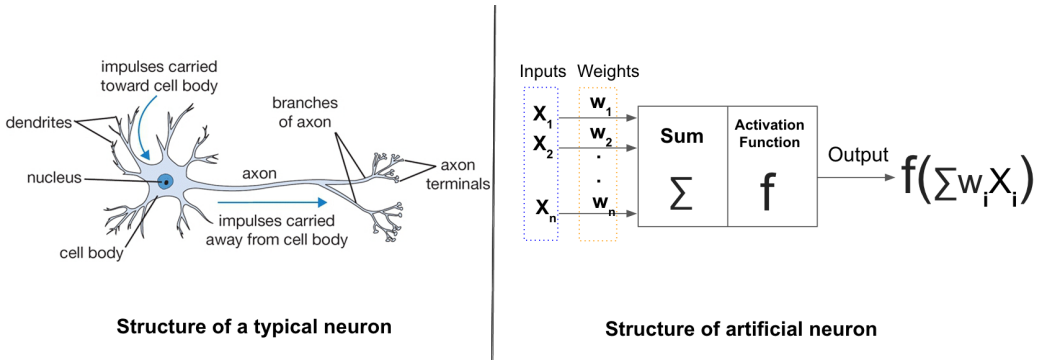

Figure 6.1: A representation of a biological neuron (left) and its mathematical model (right) [7] [5].

6.2 Feed-forward Neural Networks

Neural Network models are mostly organized and represented as distinct layers of neurons. For regular neural networks, the most common type are the fully-connected, feed-forward neural networks This kind of networks have three types of layers: input, hidden and output, where neurons between two nearest layers are pairwise connected - signal travels from the input node, through the hidden one and ends at the output layer.

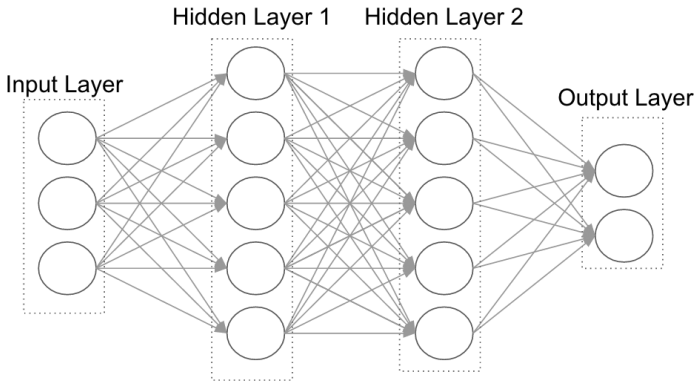

Figure 6.2: A 3-layer feed-forward neural network with three inputs, two hidden layers of 4 neurons each and one output layer [5].

6.3 Activation Functions

The main role of activation functions is to make a neural network non-linear by transforming the weighted sum of inputs that come into the neurons. The function should include nonlinearity to reflect complex structures of the data [7].

The most popular activation functions:

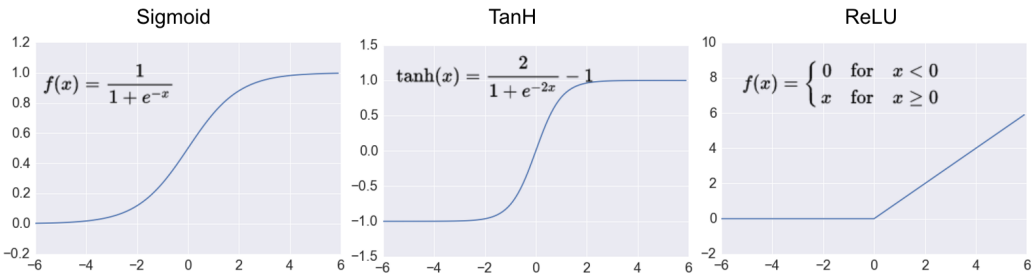

Figure 6.3: The most popular activation functions. [5]

6.4 Convolutional Neural Networks

Convolutional Neural Networks (ConvNets or CNNs) are a special type of feed-forward networks that were designed to emulate the behaviour and structure of a visual cortex of the brain. They are most suitale for visual recognition tasks.

Standard feed-networks requires connection between all neurons, resulting in too complex network structure. When a network needs to be trained on a large dataset, the cost of complexity grows (more layers needs to be added) - the more complex the network is, the more computational power is needed. ConvNets have special proprties that allows the network to encode certain properties of given datasets (such as specific details on images) by adding to the network model additional layers: **convolutional layer** and **pooling layer** [5].

The idea of CNNs can be summarized in four steps:

1. Convolution.
2. Subsampling.
3. Activation.
4. Fully connected layer.

6.4.1 Convolution

Convolution layer, the first layer that receives an input data, that consists learnable filters that try to label the input signal by referrence to what it has learned in the previous iterations. These filters are activating when they can recognise a specific, repetitive structure, which they had been able to seen before [5].

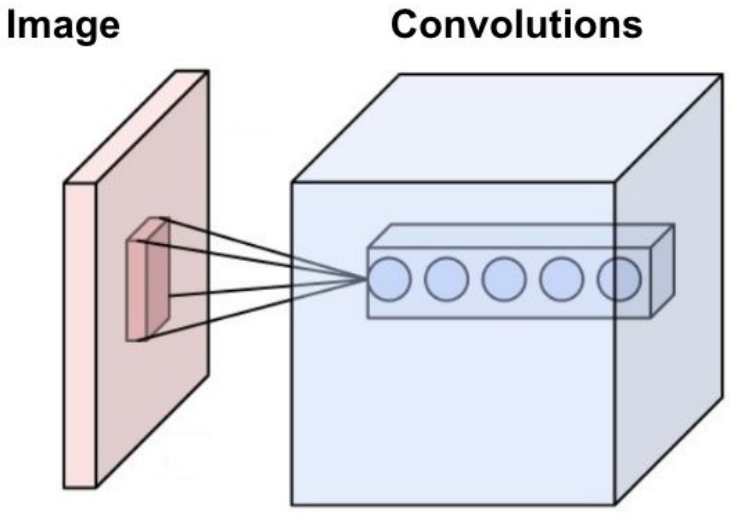

Figure 6.4: Neurons of a convolution layer [7].

6.4.2 Subsampling

The subsampling or smoothering process is the part of building a convolution network where the inputs can process convoluted data to reduce the sensitivity of the earlier used filters to all kinds of noises and variations. This process can be achieved by using averages or the maximum over a sample of the data. Examples methods: reducing the size of the image, reducing a resolution of the image, reducing the color contrast across RGB channels and other data processing methods [5].

6.4.3 Activation

Activation function controls how the signal flows between layers by using activatin fuctions [5].

6.4.4 Fully connected layer

The fully connected layer is he last layers in the network. It ensures that all neurons of preceding and subsequent layers are fully connected to each other [5].

6.5 CNNs Architecture

The simplest architecture of a CNN has an input layer (for example a set of images) which is followed by followed by convolutional (with layer of activation functions) and pooling layers; and the sequence ends with fully-connected layers.

The early stages of the sequence recognizes generic patterns, while later layers are responsible for detection of more detalic patterns [5].

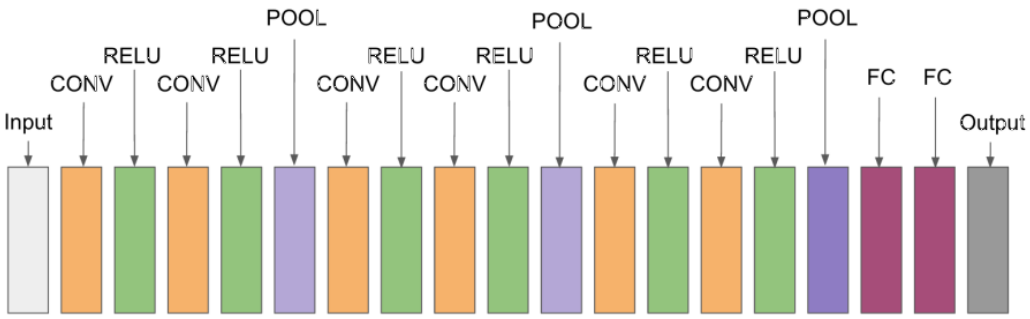

Figure 6.5: Example of a CNN architecture [7].

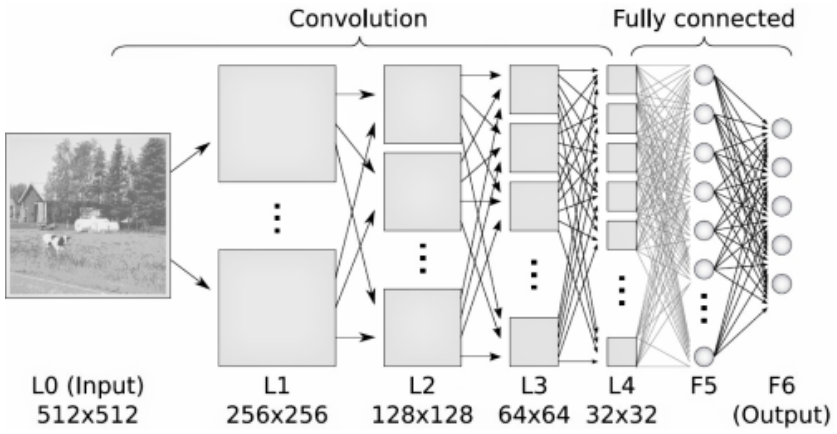

Figure 6.6: The LeNet: the most popular implementation of the CNN [7].

7

Solution and Results

In this section we will investigate the whole process of building suitable deep learning method which will be used for morphological classification of galaxies. The whole chapter explains the basic introduction to organization and preparation of data resources and construction of a neural network.

7.1 Data preprocessing

The provided data by the Galaxy Zoo was a set of 141 553 JPG images of galaxies (61 578 for a training process and 79 975 for a test process) and a CSV file which contained probability distributions for the classifications for each of the training images gained through on-line stage of the classification process. Each probability distributions are based on the answers provided by the volunteers and describe the likelihood of the galaxy falling in a specific morphological category (see the Galaxy Zoo decision tree - chapter 5.2.1) [10].

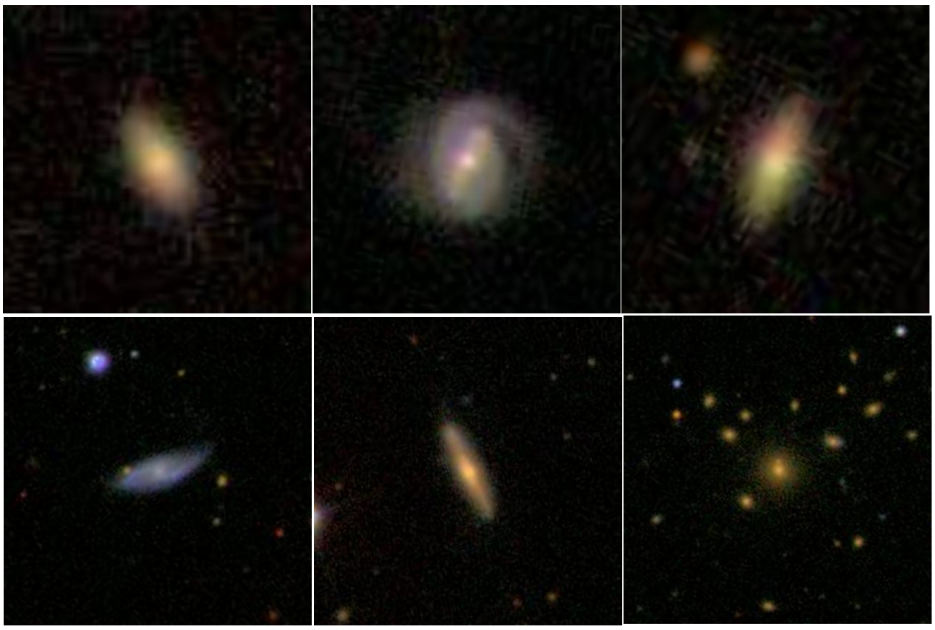

Figure 7.1: Example images provided by the Galaxy Zoo.

7.1.1 Selection of the proper images

The first stage was to choose appropriate images and put them to corresponding directories.

To simplify the approch of galaxy classification only photos of probabilities higher than 0.7 were used and by that over 6 000 images was selected in each cattegory in both training and test sets - about 70% of volunteers decided that a image that was shown to them belongs to one of the categories (such as ellipticals or spirals).

Next step was to move chosen images to adequate catalogues by dividing them into two seperate classes. Training and validation sets of the images was arranged into independent places:

```
galaxies/
    train/
        elliptical/
            elliptical1.jpg
            elliptical2.jpg
            ...
        spiral/
            spiral1.jpg
            spiral2.jpg
            ...
    validation/
        elliptical/
            elliptical1.jpg
            elliptical2.jpg
            ...
        spiral/
            spiral1.jpg
            spiral2.jpg
            ...
```

Figure 7.2: Structure of the catalogues after location proces of images of the galaxies.

7.1.2 Cropping of the images

The data consisted of colour (three color RGB channels) JPG images in resolution of 424 x 424. Almost all images contains galaxies at the center - the void of space outside the centers is not needed for the purpose of recognition of the morphologies - Thus the images was cropped at the center using 256 x 256 windows.

In order to shorten the time of the calculations the images was also resized to 128 x 128 resolution.

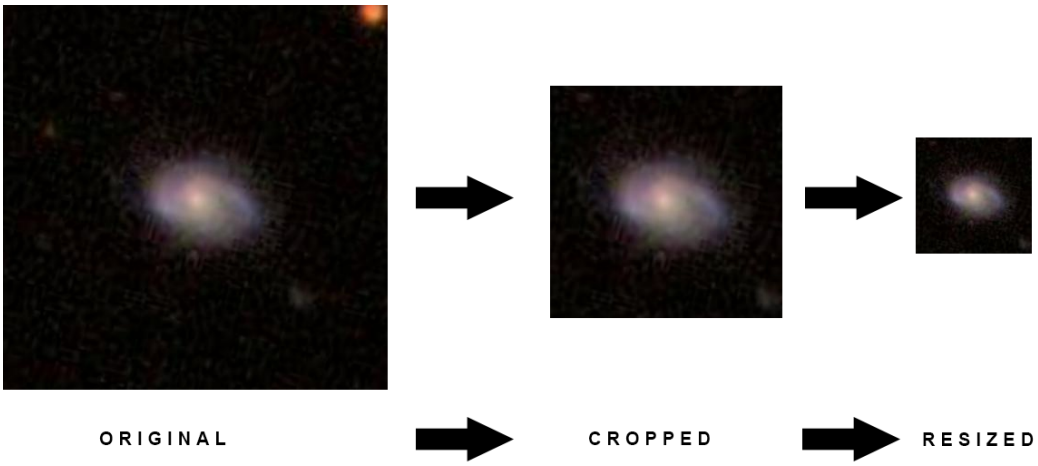

Figure 7.3: Comparission of the original and changed image.

7.2 Software and hardware

The galaxy classifier was implemented by using Python programming language and Keras library (with TensorFlow backend).

The networks were trained on Amazon Web Services (AWS) instance with Nvidia Tesla K80 GPU (4992 Nvidia CUDA processing units) and traditional workstation with Nvidia GeForce GTX 950M (640 Nvidia CUDA processing units).

7.3 Network architecture

For the purpose of ths thesis was adopted an ConvNet architecture called **VGG16** [47] - a very deep neural network architecture that have 16 weight layers.

The VGG16 model was created in 2014 by Karen Simonyan and Andrew Zisserman during their work on ImageNet Challenge 2014 Contest [15]. They proved that CNNs with 16 (or more) deep layers are more than appropriate

for image classification tasks.
Since then many researchers have tested this approach what has contributed to drastic improvements in the field of computer vision and thanks to that now we can use many pre-trained models to improve newly created algotithms.

7.3.1 Pre-trained model

Instead of creating from scratch a completely new architecture we can use a more sophisticated approach - we can use model already pre-trained on a large dataset. Such model would have already learned and recognized features that are useful for most classifications problems, and using these features would let us to reach a better precision than any other method that purely relys only on the accessible data [47].

The VGG16 architecture is perfectly suited to this kind of tasks. It is model that was pre-trained on the ImageNet dataset, dataset that contains more than 1 000 classes of images - this model already have implementations of some image properties that may be relevant to our task [25][26].

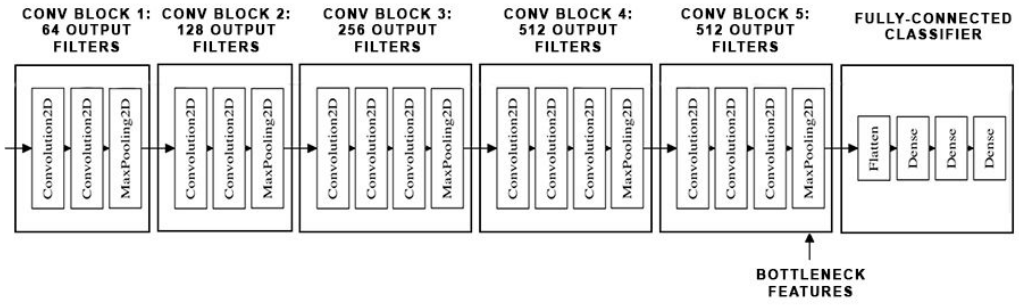

Figure 7.4: Visual representation of VGG16 architecture [3].

7.3.2 Fine-tuning the pre-trained architecture

In order to change the pre-trained network for our purpose, only the convolutional blocks of the model were instantiated without changing their content - recording the outputs from single run of the the model trained on our datasets. The important part of this step is to read the outputs only from the he last activation layer (the "bottleneck features") not from the fully-connected one. Only after that we can train fully-connected model on top of the outputs obtained in the previous step.

To improve our results, we can try to change the last convolutional block (Conv block 5) and the first layer of classifier block. We will start from a trained network, then re-training it by slightly changing weight numbers which are provided with VGG16 model , but the activation function (in this case ReLU - chapter 6.3) needs to remain unchanged [19][3].

The fine-tuning process can be achive in three steps:

1. Initiation of the base VGG16 model, with its weights.
2. Adding our trained model on top of the VGG16 (with exclusion of last block of layers).
3. Changing only the blocks after the "bottleneck features".

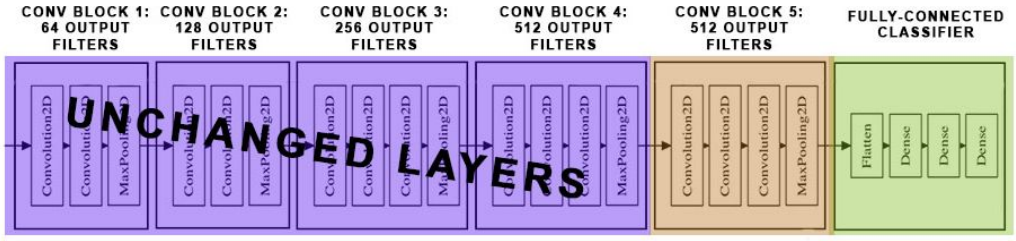

Figure 7.5: Visual representation of VGG16 fine-tunned architecture [3].

7.4 Training and results

Before we can go to the results one more concept needs to be explained - the concept of **epochs**. In process of training deep neural networks, one epoch means one pass of the training algorithm whrough the full training set. Usually it may contain a few iterations. The training algotirhm does not just go through a set once. Such process can even take thousands of iterations in order to obtain satisfactory results [37].

This approach of using pre-trained neural network model gets us to a validation accuracy of 0.86 after 200 epochs and the whole training process took about 6 hours, taking into account the fact that each training sets contained over 6 000 dark and fuzzy images. For comparison, three years ago state of the art in image classifications algorithms was 80% [16] and todays models can achive accuracy of over 90% [19].

The best way to visualize the performance of the algorithm is confusion matrix (error matrix). The confusion matrix presents how well an algorithm worked on validation part of the images [27].

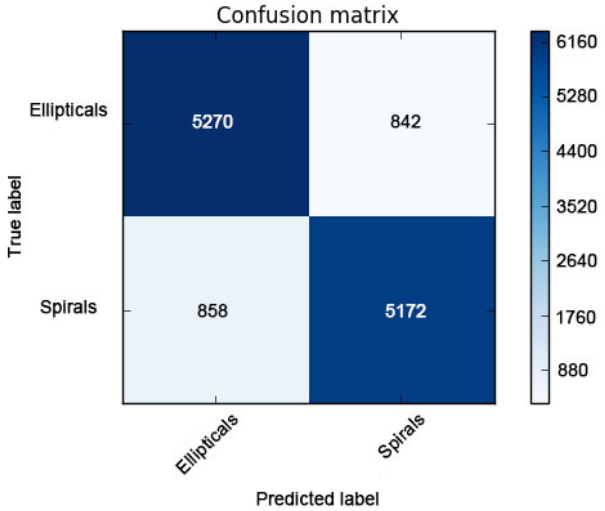

Figure 7.6: The confusion matrix for the image classifier.

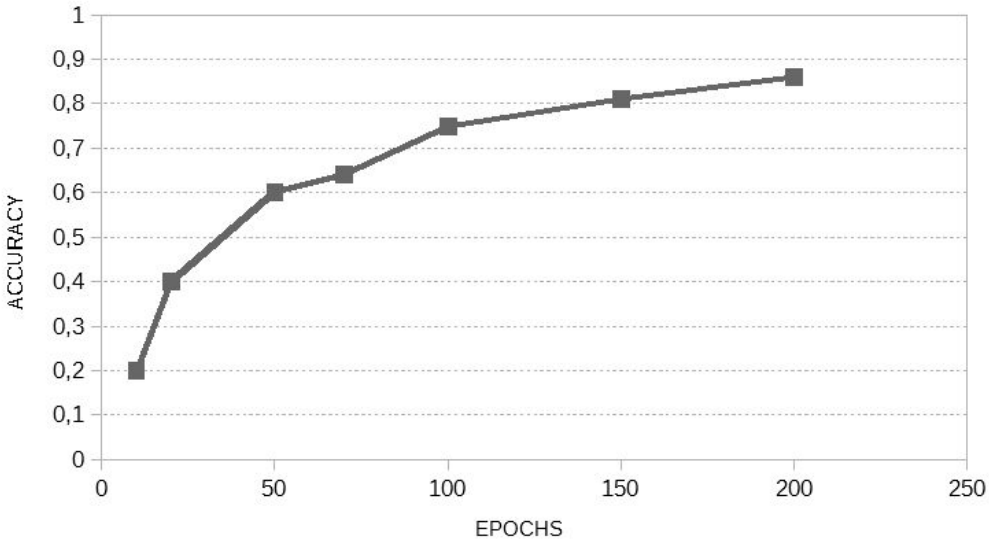

Figure 7.7: Chart that illustrate increasing accuracy of the model.

Visualisation problem	Accuracy
The galaxy classification	0.86
Invasive Species Monitoring - Kaggle challenge (2016-2017)	~0.92 (Average of all entries)
Machine Learning Attacks Against the Asirra CAPTCHA (2014) by Philippe Golle	0.80
Deeply-Supervised Nets (2014)	0.91
Committees of deep feedforward networks trained with few data (2014)	0.68
Deep Learning of Invariant Features via Simulated Fixations in Video (2012)	0.56

Figure 7.8: Comparative table which presents the results of the galaxy classifier in comparision to other visual recognision works [8].

8

Conclusions

On the pages of this thesis the author clarified the analogy between Artificial Neural Networks and the human brain, described and introduced basic theory behind Galaxy Morphological Classification system as well as the methods of Artificial Intelligence and their influence the lives of each of us.

Deep Learning Algorithms and especially Convolutional Neural Network many times have demonstrated their outstanding capabilities in many areas of life and yet again, the metods have proven their worth in such difficult field that is Morphological Classification of Galaxies. The success rate of 86% on the testing dataset of more than few thousands images of galaxies speaks by itself and shows that it is possible to develop an efficient neural network-based algorithms capable of reliable object type classification based on data delivered by The Galaxy Zoo Project. If the future appears more detailed images of the sky it will be possible to improve accuracy and predictions rates of Machine Learning methods used in astronomical studies.

9

Source code

This chapter contains basic CovNet elements written in Python, along with comments and concise explanations, which was used for the purpose of this thesis.

Importing basic libraries and dependencies:

```
import classes
import models
from keras.preprocessing.image import ImageDataGenerator
from models.Base import CropScaleImageTransformer
from keras import optimizers
from keras.preprocessing.image import array_to_img,
from keras.preprocessing.image import ImageDataGenerator
from keras.preprocessing.image import img_to_array
from keras.preprocessing.image import load_img
from keras import applications
from vgg16 import VGG16
from keras.models import Sequential
from keras.layers import Dropout
from keras.layers import Flatte
from keras.layers import Dense
import time
import csv
```

```
import numpy as np
```

Function that crops and rescale images:

```
def get_images(c=256, s=128):

    """
    Crop: 256x256
    Rescale: 128x128
    """

    cs = CropScaleImageTransformer(training=True,
                                   result_path='data/img_end.npy'
                                   .format(crop, s),
                                   crop_size=c,
                                   scaled_size=s,
                                   memmap=True
                                   n_jobs=-1,
                                   )
    imag = cs.transform()
    return imag
```

Importing VGG16 architecture and weights [25][26] of the top layers, as well as setting up directories and numbers of training images:

```
weights_vgg = '../vgg16_weights.h5'
top_model_vgg = 'fc_model.h5'

# resolutions of the images:
img_width = 150
img_height = 150
```

```
train__dir = 'galaxy/train'
test__dir = 'galaxy/test'

train_samples = 6125
test_samples = 6075

epochs = 2000
batch_number = 1000
```

Initiating the VGG16 architecture:

```
model = applications.VGG16(weights='imagenet', include_top=False)
```

Initiating a classifier layers - the layers on top of the convolutional model:

```
classr = Sequential()
classr.add(Flatten(input_shape=model.output_shape[1:]))

"""
Using the ReLU activation function
"""
classr.add(Dense(256, activation='relu'))
classr.add(Dropout(0.5))
classr.add(Dense(1, activation='relu'))

classr.load_weights(top_model_vgg)
```

Adding the the convolutional layers on top of the rest layers:

```
model.add(classr)
```

Blocking (freezing) the first layers and compiling the current model:

```
for layer in model.layers[:15]:
    layer.trainable = False
```

```
# setting slow learing speed:
model.compile(loss='binary_crossentropy',
              optimizer=optimizers.SGD(lr=1e-3, momentum=0.5),
              metrics=['accuracy'])
```

Fine-tuning the pre-trained model:

```
model.fit_generator(

    t_generator,
    samples_per_epoch=train_samples,
    epochs=epochs,
    validation_data=validation_gen,
    nb_val_samples=test_samples)
```

Data augmentation and preparation, testing results:

```
train_data = ImageDataGenerator(
    rescale= 1./255,

    shear_range=0.3,
    zoom_range=0.3,

    horizontal_flip=False)

test_data = ImageDataGenerator(rescale= 1./255)
```

Data validation:

```
test_gen = train_datagen.flow_from_directory(
    train_dir,
    target_size=(img_height, img_width),
    batch_size=batch_size,
```

```
    class_mode='binary')

validation_gen = test_datagen.flow_from_directory(
    test__dir,
    target_size=(img_height, img_width),
    batch_size=batch_number,

    class_mode='binary')
```

Bibliography

[1] Ada lovelace biography. https://www.biography.com/people/ada-lovelace-20825323.

[2] A black hole's dinner is fast approaching. https://www.eso.org/public/news/eso1151/.

[3] Building powerful image classification models using very little data. https://blog.keras.io.

[4] The candels official website. http://candels.ucolick.org.

[5] Convolutional neural network (cnn) introduction. https://algobeans.com/2016/01/26/introduction-to-convolutional-neural-network.

[6] The cosmic distance scale. https://imagine.gsfc.nasa.gov/features/cosmic/milkyway_info.html.

[7] Cs231n: Convolutional neural networks for visual recognition. http://cs231n.github.io/neural-networks-1.

[8] Discover the current state of the art in objects classification. http://rodrigob.github.io/are_we_there_yet/build/classification_datasets_results.html.

[9] Forbes: Machine learning is redefining the enterprise in 2016. https://www.forbes.com/sites/louiscolumbus.

[10] Galaxy zoo - the galaxy challenge. https://www.kaggle.com/c/galaxy-zoo-the-galaxy-challenge.

[11] Galaxy zoo official website. http://www.galaxyzoo.org.

[12] Geoffrey e. hinton: Official website. http://www.cs.toronto.edu/~hinton.

[13] Kaggle: Galaxy zoo - the galaxy challenge. https://www.kaggle.com/c/galaxy-zoo-the-galaxy-challenge.

[14] Keras official website. https://keras.io.

[15] Large scale visual recognition challenge 2014 (ilsvrc2014). http://www.image-net.org/challenges/LSVRC/2014.

[16] Machine learning attacks against the asirra captcha. http://xenon.stanford.edu/~pgolle/papers/dogcat.pdf.

[17] Numpy official website. http://www.numpy.org.

[18] Phil armitage - astrophysics lectures, university of colorado. http://jila.colorado.edu/~pja/astr3830/.

[19] Practical deep learning for coders - lessons 2-4. http://course.fast.ai/.

[20] Python official website. https://www.python.org.

[21] The skyserver official website. http://www.skyserver.org.

[22] The sloan digital sky survey official website. http://www.sdss.org.

[23] Super massive black hole in the center of the milky way - esocast. https://youtu.be/A18jCuZCh84.

[24] Tensorflow official website. https://www.tensorflow.org.

[25] Vgg16 pre-trained model for keras 1. https://gist.github.com/baraldilorenzo/07d7802847aaad0a35d3.

[26] Vgg16 pre-trained model for keras 1. https://gist.github.com/ksimonyan/211839e770f7b538e2d8#file-readme-md.

[27] Wikipedia: Confusion matrix. https://en.wikipedia.org/wiki/Confusion_matrix.

[28] Wikipedia: Cuda. https://en.wikipedia.org/wiki/CUDA.

[29] Wikipedia: Hubble sequence. https://en.wikipedia.org/wiki/Hubble_sequence.

[30] Wikipedia: Laniakea supercluster. https://en.wikipedia.org/wiki/Laniakea_Supercluster.

[31] Wikipedia: Machine learning. https://en.wikipedia.org/wiki/Machine_learning.

[32] Wikipedia: Sloan digital sky survey. https://en.wikipedia.org/wiki/Sloan_Digital_Sky_Survey.

[33] Wired: Meet the man google hired to make ai a reality. https://www.wired.com/2014/01/geoffrey-hinton-deep-learning.

[34] Aleks Buczkowski. What's the difference between artificial intelligence, machine learning and deep learning? http://geoawesomeness.com/whats-difference.

[35] Ronald J. Buta. Galaxy morphology. https://ned.ipac.caltech.edu/level5/Sept11/Buta/frames.html, Online, accessed 2011.

[36] Michael Copeland. Nvidia blog: What's the difference between artificial intelligence, machine learning, and deep learning? https://blogs.nvidia.com/blog/2016/07/29/.

[37] Yoshua Bengio Ian Goodfellow and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[38] J. Richard Gott Mario Juric. A map of the universe. *arXiv, astro-ph.co*, 0310571v2, 2005.

[39] Chris J. Lintott Kyle W. Willett. Galaxy zoo 2: detailed morphological classifications for 304,122 galaxies from the sloan digital sky survey. *arXiv, astro-ph.co*, 308.3496v2, 2013.

[40] Raquel Yumi Shida Lars Lindberg Christensen, Davide de Martin. *Cosmic Collisions - The Hubble Atlas of Merging*. Springer, 2009.

[41] Ofer Lahav Manda Banerji. Galaxy zoo: Reproducing galaxy morphologies via machine learning. *arXiv, astro-ph.co*, 908.2033v2, 2010.

[42] Calum Mclelland. The difference between artificial intelligence, machine learning, and deep learning. `https://www.leverege.com/blogpost`.

[43] Adil Moujahid. A practical introduction to deep learning with caffe and python. `http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe`.

[44] Hope Reese. Understanding the differences between ai, machine learning, and deep learning. `http://www.techrepublic.com/article/understanding-the-differences-between`.

[45] Anuraj Mohan Remya G. Deep learning approach for classifying galaxies. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2277 128X, 2016.

[46] James Schombert. Astronomy 123: Galaxies and the expanding universe: Hubble sequence. `http://abyss.uoregon.edu/~js/ast123/lectures/lec11.html`.

[47] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 0310571v2, 1409.1556.