Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Master's Thesis

Samim Ahmad Multaheb

Transfer Possibilities of a Deep Learning System from Medicine into Aviation

Fakultät Technik und Informatik
Department Maschinenbau und Produktion

Faculty of Engineering and Computer Science
Department of Mechanical Engineering and Production Management

Samim Ahmad Multaheb

Transfer Possibilities of a Deep Learning System from Medicine into Aviation

Master's thesis submitted as part of the Master's examination

in the study program Production Engineering and -Management
at the Department of Mechanical Engineering and Production
of the Faculty of Engineering and Computer Science
of the Hamburg University of Applied Sciences

in cooperation with:

Lufthansa Technik AG
Original Equipment Innovation
Weg beim Jäger 193
22335 Hamburg

Universitätsklinikum Hamburg-Eppendorf
Zentrum für Molekulare Neurobiologie Hamburg
Institut für Medizinische Systembiologie
Falkenried 94
20251 Hamburg

Primary examiner: Prof. Dr.-Ing. habil. Frank Helmut Schäfer
Secondary examiner: Prof. Dr. rer. nat. Stefan Bonn
Industrial examiner: Frank Niss and Franziska Voerner

Date of submission: 11.05.2018

Abstract

Name of Student

Samim Ahmad Multaheb

Title of the paper

Transfer Possibilities of a Deep Learning System from Medicine into Aviation

Keywords

Deep learning, Machine learning, Artificial intelligence, Use cases from medicine, Use cases from aviation, Technology transfer, Transfer bridge, Early stage technology development, Exon detection, Damage detection, Lightning strike damage, Object detection, Natural language processing, Speech recognition, Learning algorithms

Abstract

This thesis deals with the possibilities to learn from medicine how to apply deep learning technology in aviation. From the basic understanding of artificial intelligence (AI), machine learning, deep learning, and the precise distinction between these fields, to the advantages and limitations of deep learning, this thesis aims to give an introduction to the complex field of AI and deep learning. The theory is underlined with various deep learning use cases in both industries. The transfer bridge between two similar use cases from medicine and aviation, genomic exon detection and lightning strike damage detection is built methodologically to investigate possible profits of the transfer from academia to the private sector.

Name des Studenten

Samim Ahmad Multaheb

Thema der Masterarbeit

Transfermöglichkeiten eines Deep Learning Systems aus der Medizin in die Luftfahrt

Stichworte

Deep Learning, Machine Learning, Künstliche Intelligenz, Use Cases aus der Medizin, Use Cases aus der Luftfahrt, Technologietransfer, Transferbrücke, Technologieentwicklung im Frühstadium, Exon-Detektion, Schadensdetektion, Blitzschlagschaden, Objektdetektion, Natural Language Processing, Spracherkennung, Lernende Algorithmen

Kurzzusammenfassung

Diese Arbeit umfasst die Möglichkeiten von der Medizinbranche zu lernen wie die Deep Learning Technologie in der Luftfahrt eingesetzt werden kann. Vom grundlegenden Verständnis der Künstlichen Intelligenz (KI), Machine Learning und Deep Learning, sowie der präzisen Unterscheidung zwischen diesen Feldern, bis zu den Vorteilen und Limitierungen von Deep Learning, versucht diese Arbeit einen Einstieg in das komplexe Feld der KI und Deep Learning zu ermöglichen. Die Theorie wird mit verschiedenen Use Cases aus beiden Industrien betont. Die Transferbrücke zwischen zwei ähnlichen Use Cases aus der Medizin und Luftfahrt, Exon-Detektion aus der Genomik sowie Blitzschlag-Schadensdetektion ist methodisch aufgebaut um mögliche Profite aus dem Transfer von der Wissenschaft in den kommerziellen Sektor zu untersuchen.

Index

List of Figures

List of Tables

Abstract

Although artificial intelligence has had its neuroscientific beginning with artificial neurons more than three quarters of a decade ago, it still remains a promising research field with highly anticipated applications. Deep learning in particular has shown rapid development over the course of the last five years.

This thesis deals with the advantages of this progressive technology using deep neural networks over traditional machine learning approaches. In addition to an introduction to artificial intelligence, common machine learning algorithms were described for a better understanding. The main focus lied on the transfer of technology not only from academia to the private sector, but also from the medical industry to aviation. Given the background in aviation and engineering, the problems regarding the aviation industry were known. Moreover, methods of problem understanding were used on-site in the maintenance shop based on Toyota's model of *genchi genbutsu* to understand the problems profoundly. On top, creativity sessions helped detect novel use cases worth considering. The use case of autonomous visual inspection with the help of deep learning, referred to as AutoInspect, was described extensively in this thesis, alongside with the similar medical use case of exon detection to build transfer links. The research subject was detecting damages, in particular lightning strike damages to the aircraft's skin, with deep learning. Additionally, the preprocessing was done for one example of lightning strike damage to show how an image input to a machine optimally should look like. From the findings, it was derived that medicine and aviation have a lot in common when it comes to implementing a powerful AI technology. Throughout the aircraft's and patient's journey, there are many docking points that allows deep learning to enhance services, predict states and fulfill tasks that are difficult for a human or traditional machine learning approaches, and derive valuable knowledge from data in a dimension that was impossible until now.

The thesis ends with the description of future research topics of the AutoInspect concept, which includes an autonomous unmanned aerial vehicle, online detection function, and a mapping of the findings on a three-dimensional aircraft map. Of further research interest is also the integration of a deep learning system in a complex real-world environment like the maintenance hangar, and a standardized process for high-quality data acquisition to relieve the bottleneck of deep learning.

1 Introduction

Artificial intelligence (AI) is seen as one of the forward-looking technologies of the next ten years. Despite that since several decades there are efforts to intelligently design a machine, the issue still remains interesting and seems to create added value in action with other promising technologies. While applications like chat bots and translation tools are already in use today, other fields like predictive analytics and general machine intelligence are still topics of the future. Exactly the latter seem to be very interesting for industries like general mechanical engineering and production, automotive, aviation and other focal subjects like industry 4.0, machine communication and piloted driving.

In the 2017 Gartner hype cycle for emerging technologies (fig. 1.1), the extensive implementation of machine intelligence in autonomous vehicles for example, is expected for the coming ten years (Panetta, 2017). Machine learning was at the climax of the hype curve, the *peak of inflated expectations*, in 2016 and one year later still remains near that point, with just *deep learning* topping it as the most hyped technology for now. However, it is expected that there will be different difficulties in implementing these future technologies. Through joint research of industry and science as well as committee work and consortia like the 5G Automotive Association, foreseeable obstacles like data transfer are already being tackled today (Voigt, 2017). It is hoped that key findings of the transfer of already developed technology from medicine to aviation makes this process less risky.

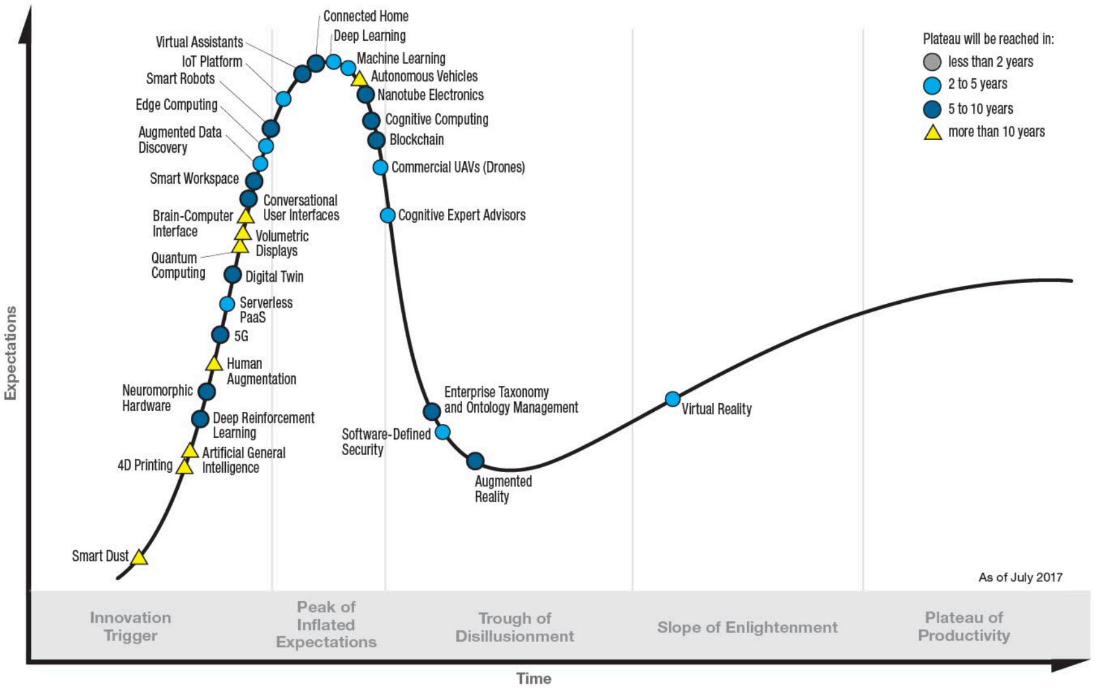

Fig. 1.1: Gartner hype cycle for emerging technologies 2017 (Gartner, 2017)

As Gartner states, Artificial intelligence is viewed as one of three mega trends, apart from digital platforms and transparently immersive experiences. It is said that executives and technology leaders should explore these three mega trends to evaluate future impacts to their business and get competitive advantages (Panetta, 2017). AI as such augments several present technologies and enables much more opportunities for the future. As of today, computers outperform humans in a large number of calculation-wise operations. The goal of AI is to expand this superiority to more complex tasks like image recognition (machine vision), extracting the essence of a certain text (text mining), generating vast knowledge databases (expert systems) and the ability to learn and to adapt to new environments (machine learning).

1.1 Aim of this Thesis

Technology transfer is a highly potent way for both academia as well as the private sector to derive profits from scientific findings, and establish business and competitive advantages. The goal of this thesis is to analyze and point out possible opportunities to transfer the technology of deep learning from medicine into aviation. It is known that deep learning is not derived from medicine originally but used in medical fields. In addition to the aim of analyzing existent use cases that use deep learning in both medicine and aviation, it is also an objective to find one specific use case to analyze deeper and point out transfer links. Of importance is also, that this use case is viable from an engineering and financial perspective. To be able to do that, an extensive on-site approach is inevitable, in both the medical and aviation industry, working jointly at facilities of both industries which are introduced in subchapter 1.3. It is hoped that both entities support this thesis in all ways possible.

Since medicine and aviation are not close relatives in regard of used technologies, deep learning might be a common thread. The analogy of the patient and aircraft, where limbs and components, respectively, are in need of professional analysis to maintain the system, plays an important role. Both *patients* may have diseases or failures that need to be cured before the next step is to understand the disease itself. In addition to aircraft components (predictive maintenance) and the aircraft itself (skin deficiency or damages), this might also be applicable to use cases inside the cabin or the cockpit, where deep learning could be used to analyze passenger's or pilot's behavior, just as it can be used for predictive diagnosis or mental health analysis of patients in the medical field.

The thesis itself is not to be understood as a standalone analysis. It is a personal goal to further deepen my expertise in the fields of deep learning, interdisciplinary work and problem solution. It is understood, that this paper might end in a state that needs a managerial decision to take actions for the next steps. It is desired on my behalf to be part of these next steps to develop and implement deep learning in an aviation environment.

1.2 Structure of this Thesis

A rough outline is visualized in Fig. 1.2. It is aimed that the outcome of this thesis builds a foundation for further research in the direction of interdisciplinary learning and the application of the learnt, with regard to artificial intelligence and deep learning.

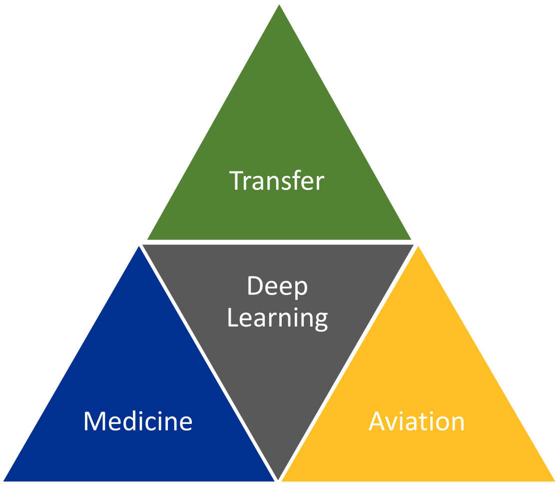

Fig. 1.2: Rough outline of the covered fields in this thesis

Besides the introduction to artificial intelligence, the historical development and the definition and types of intelligence, the basic chapter further deals with the nature of learning algorithms and aims to give a basic understanding of machine learning and deep learning, as well as the distinction between both fields (Fig. 1.3). The main part is then split into an analyzing part where it will be looked into the medical field as well as aviation, deriving use cases with deep learning applications, and a solution-oriented part, dealing with the possibilities of technology transfer from one industry to another. To be able to make statements to the possibilities of transfer, the outline process is analyzed and connected with the specific main use cases of medicine and aviation. Also, the findings might lead to hidden opportunities and results that were not foreseeable at the time writing. At last, the findings will be collated and a recommendation for action will be given, in addition to the conclusion of the analyzed matters.

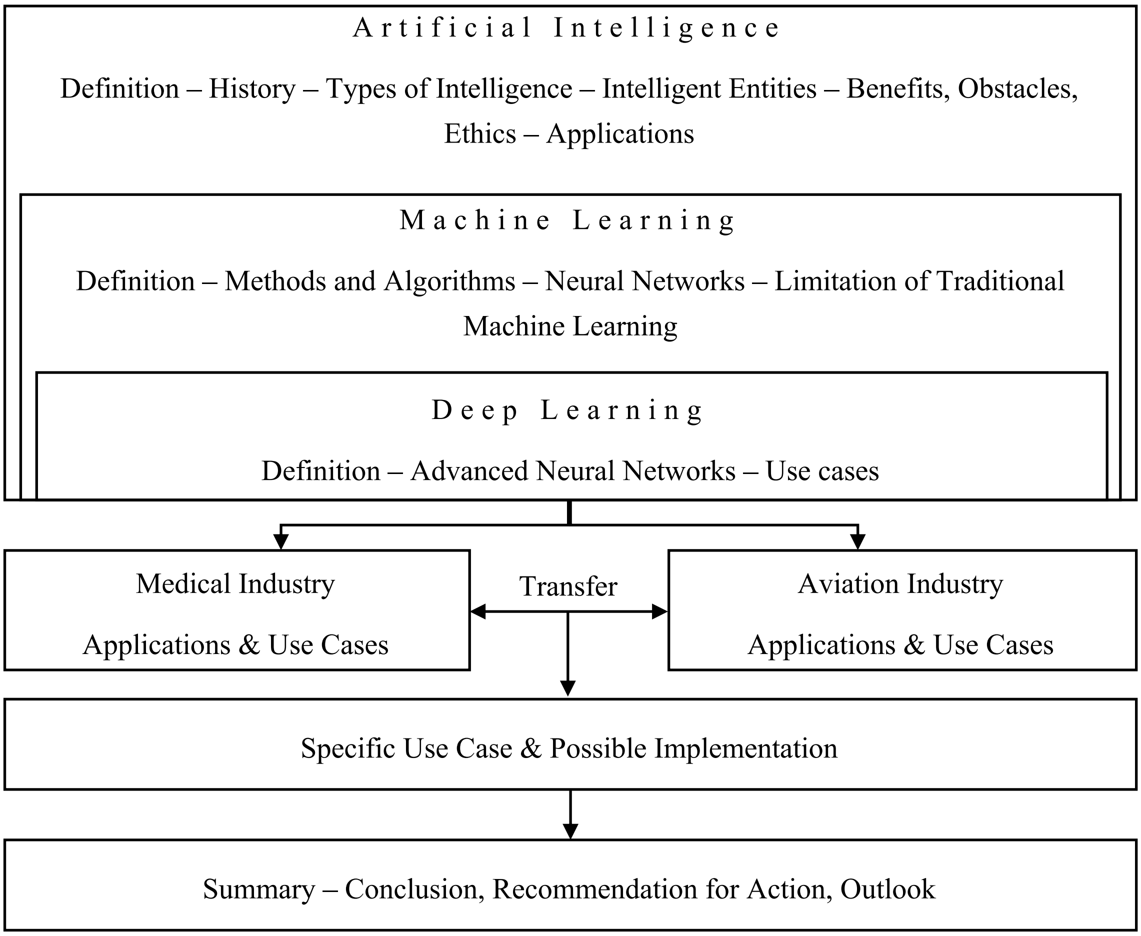

Fig. 1.3: Structure of this thesis

1.3 Background

The present thesis was prepared and written in an approach to both analyze the medical sector as well as aviation. For the purpose of getting a better understanding, a brief introduction to both the University Medical Center Hamburg-Eppendorf (UKE) and Lufthansa as well as Lufthansa Technik (LHT), will be given. Both Lufthansa Technik and the UKE were key supporters of this thesis, exceptionally Frank Niss and Prof. Dr. Stefan Bonn as well as Prof. Dr.-Ing. habil. Frank-Helmut Schäfer from the University of Applied Sciences Hamburg who mentored this thesis from an industrial and academic point of view. I also am thankful to the colleagues from Lufthansa Technik, especially in Hamburg and Frankfurt, as well as the scientists and doctoral candidates from the UKE, that helped me in the process of writing this thesis and combined findings from the medical and the aviation industry.

My background is, graduating from a University of Applied Sciences, a more hands-on approach. This includes methodological analysis of problems, deriving of problem solutions, viability and feasibility analysis as well as engineering and technology management. Of my personal interests are early stage technologies. This was met by the topic of this thesis, analyzing a cutting-edge artificial intelligence technology for the use in a real-world environment. In terms of studies, the focus was intentionally set apart of the classical mechanical engineering fields to broaden the horizon and learn from more inputs with the aim to be ready for the challenges of Industry 4.0 and digitalization.

1.3.1 University Medical Center Hamburg-Eppendorf

The University Medical Center Hamburg-Eppendorf, one of the largest hospitals in the Hamburg area and one of the most modern in Europe, was founded 1889 and became a university medical center in 1934. In 14 centers, it comprises more than 80 interdisciplinary clinics and institutes, and employs more than 9,000 people, including 2,300 doctors. Approximately 86,000 inpatients and 269,000 outpatients, as well as 50,000 emergency cases are being treated every year. Apart from medical treatment, science and research at the University Medical Center Hamburg-Eppendorf enjoys an international reputation. Interdisciplinary research is also done in the field of augmentation through artificial intelligence to further improve service and medical treatment.

1.3.1.1 Center for Molecular Neurobiology Hamburg

Established as the first center in Germany for basic research in molecular neurobiology in 1987, the Center for Molecular Neurobiology Hamburg is divided into five institutes, eight research groups, two guest groups and seven core facilities. The main focus lies in the research of neural diseases and developing new treatment methods. Moreover, the Center issues pioneering research publications in high-ranked peer-reviewed journals.

1.3.1.2 Institute of Medical Systems Biology

The Institute of Medical Systems Biology is the latest establishment of the Center for Molecular Neurobiology Hamburg. The institutes goal is focused on understanding human pathology, especially in the central nervous system. In an age where information exists abundantly, it is crucial to derive knowledge from this data, and thus use it to enhance medical treatment. This approach is based mainly on artificial intelligence techniques such as deep learning, which will be analyzed in this thesis. The institute uses deep learning amongst other things for object detection and so called long short-term memory cells in genomics, and generative models to simulate data. Moreover, it is thought that the technologies used in the institute is universally applicable in other environments and industries. The institute’s director Prof. Dr. Stefan Bonn acts as a supervisor to this thesis.

1.3.2 Lufthansa Group

The Lufthansa Group is a globally operating air carrier with its headquarters in Cologne. Currently about 500 subsidiary and associate companies are part of the Lufthansa Group.

The total number of employees of its five business divisions *Passage Airline*, *Logistics*, *Technic*, *Catering* and *IT-Service* count approximately 123,000. The passenger airline part *Passage* contributes to about 75% of the group revenue. However, the highly competitive air traffic market requires additional services, that mostly are covered by the subsidiaries within the Lufthansa Group. Regarding artificial intelligence, the Lufthansa Group deals with present problems throughout the passenger's journey and experience, and already uses AI to improve their services and boost up revenues. Further implementations of AI use cases are screened thoroughly inside the different branches of the group.

1.3.2.1 Lufthansa Technik AG

One of the subsidiary companies of the Lufthansa Group is the Lufthansa Technik AG. Founded 1994 as an independent company from the technical branch of the group, the headquarters with its approximately 20,000 employees, is located at the Hamburg Airport Helmut Schmidt. The Lufthansa Technik is divided into eight product divisions (PD):

- Aircraft Maintenance
- Aircraft Overhaul
- Components
- Engines
- Landing Gears
- VIP & Executive Jet Solutions
- Original Equipment Innovation
- Digital Fleet Solutions

The main focus lies in the maintenance, repair and overhaul business of aircrafts, though the demand for special needs solutions for VIP or digitalization impels the Lufthansa Technik to broaden its offerings. As an approved EASA Part 21J and G aerospace development company and manufacturing plant, as well as an EASA/FAR Part 145-approved maintenance organization, the Lufthansa Technik covers a major area of services for aircrafts and aircraft components. The newly established product division Digital Fleets Solutions deals with the coming digitalization in aviation and develops solutions for future applications in aviation. Part of it is the evaluation of the use artificial intelligence inside and outside the aircraft.

Besides Hamburg, there are few other locations in Frankfurt, Munich and Berlin. In addition, about 31 international located technical maintenance plants are part of the Lufthansa Technik.

1.3.2.2 Original Equipment Innovation

Since 2014 the product division *Original Equipment Innovation* (OEI) deals with innovation inside the aircraft cabin. The PD is further split into the four business areas

Commercial Aviation, *Business Aviation*, *Lighting* and *Seating & Structures*. Primary objective is the development of innovative products for the cabin. Those range from luminescent guiding strips and signs, induction cooking platforms and modular business seats to antenna radomes and in-flight entertainment systems. Moreover, in two staff departments (*Foresight & Insights* and *Product Planning & Development*) technology research and new product development take place to ensure sustainable technology and business development as well as new market opportunities. Foresight & Insights, in particular Frank Niss, took charge of the industry mentoring throughout the completion of this thesis.

1.3.2.3 Digital Fleet Solutions

The 2016 established PD Digital Fleet Solutions pursues the goal of digitalizing the aircraft and the processes connected throughout the lifecycle of an aircraft. Officially founded in fall 2017, the product division still in development. The possible use cases are mostly in the initial phase. In particular, and augmented through AI, there are use cases for reliability management, aircraft and aircraft health monitoring as well as predictive maintenance that use a machine learning approach. Moreover, the division is growing not just numerically but also in fields of expertise. There are however no signs of any use case using deep learning technology at the date of beginning this thesis. The division remained helpful throughout the whole process of this thesis, in particular Dr. Nima Barraci.

2 Artificial Intelligence

The *homo sapiens* (Lat. wise man) differs in one decisive respect from other mammals: He shows several features of intelligence; thus, his mental performance shapes his being significantly. The term *intelligence* is mostly used in a psychological context and stands for the cognitive capability of the human being. In philosophy, Descartes' famous statement *cogito, ergo sum* (Lat. I think, therefor I am) underlines the raison d'être of the *wise* human species (Descartes, 1870). In the area of artificial intelligence, unlike in psychology and philosophy, researchers try not only to understand intelligence, but to build and construct intelligent entities (Russell & Norvig, 1995, p. 3). According to the Oxford dictionary, artificial intelligence is defined as the *capacity of computers or other machines to exhibit or simulate intelligent behavior (and) the field of study concerned with this*. In detail, artificial intelligence enables visual perception, speech and voice recognition, decision making, translating between languages, problem solving, planning and learning, as well as manipulation and much more to machines (Techopedia, kein Datum). The current aim is to automate this process of learning to achieve tasks through learning algorithms, and to increase the technology value.

In the media, voices are being raised by critical scientists and AI-entrepreneurs that classify artificial intelligence to be a threat to humanity. Precisely because of the exaggerated portrayal of scenarios that show the extinction of the human race, if the fear comes true, the horizon of expands. This is needed to be aware of possible unwanted outcomes and to further research preventive actions. This point of thought is backed by the father of science fiction Jules Verne's statement, that the scientific progress lives on exaggerated expectations (Reusch, 1993).

This chapter discusses the historical development of intelligent entities, the basic differences between humans and machines, as well as the particular fields and instances of artificial intelligence.

2.1 Historical Development

In ancient times, narratives about non-human apparatuses and robots were told, that showed a certain intelligent behavior. The stories however were far away of breaking out of the fantasy. The first technically specific model was Da Vinci's own developed and built humanoid robot in 1495 (Fig. 2.1).

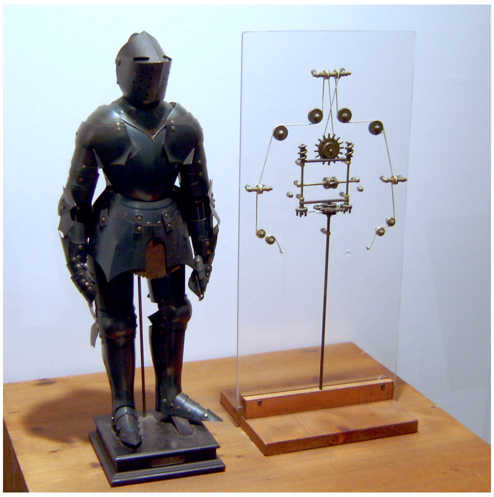

Fig. 2.1: Model of Da Vinci's humanoid robot with mechanisms inside (Da Vinci, 2005)

The first connections between neurology, information technology and cybernetics were developed in the 1940s and 50s (Bolonkin, 2011, p. 143). At that time, the research findings in neurology by McCulloch and Pitts showed, that the human brain is a complex electrical network of impuls transmitting neurons, where the neurons are characterized by the digital scale of *on* and *off*, with off being the default status and switching to on occurring in response to stimulation by a sufficient number of neighboring neurons (Russell & Norvig, 1995, p. 16). Russel and Norvig also state, that the work of McCulloch and Pitts initiated further research in this field. In the beginning of the 50s, Shannon and Turing independently developed a chess computer, just like Minsky and Edmonds developed the first computer-based neural network with forty neurons (ibid., p. 16 f.). Despite partially pessimistic stance of many academics, the boundaries of artificial intelligence were further expanded step by step. For example, in 1950 Turing drew up a list with things a machine can never do (*A machine can never do X*). In this list he included *being friendly*, *having a sense of humor*, *distinguish between right and false*, *make errors*, *get someone to fall in love with it* (the machine), and more (Blackmore, 2010, p. 281). As of today, many of these points still cannot be realized by a computer or a machine. However, pointing out specifically the issue of falling in love with a machine, with the current global integration of social networks as well as advanced progress in text and speech generation, this is not to be deemed an impossibility.

Until the 1970s, most of the knowledge of artificial intelligence was concentrated at the MIT, as well as Carnegie Mullen University, Stanford University and IBM. The enthusiastic work of young students was one of the reasons the research was further driven forward. Besides logical programs and tables, there were efforts to program all mathematical problems and let a machine solve it (Russell & Norvig, 1995, p. 17ff). One example is Evan's ANALOGY program in 1963, which could solve a simple logical problem (Fig. 2.2).

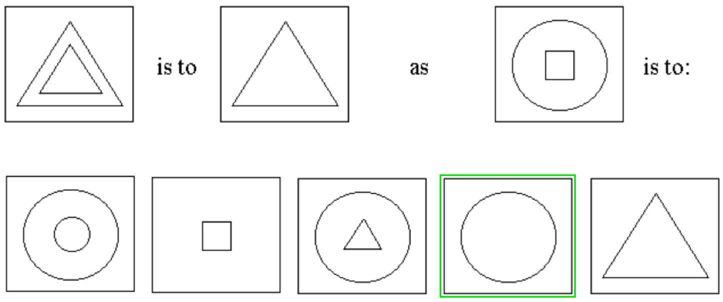

Fig. 2.2: Evan's ANALOGY program for solving analog-geometrical problems (the solution is framed green) (Smith, 2016)

Another successful example is the first chat bot in 1964, where a simulated dialog between a psychologist and his patient, who was the user, took place. In reality, the alleged psychologist was ELIZA, a program written to imitate a human academic. Most of the users, deprived of the fact that they are communicating with a machine, did not question the whole procedure, which shocked Weizenbaum, the program's developer, to an extent, that he criticized further research and development of artificial intelligence. In his opinion, a machine should never make a decision regarding humans (Smith, 2016).

After knowledge-based systems were developed in the end of the 1970s, the use of artificial intelligence in the 80s led to the development and combination of computers and the intelligent design of circuit boards to a concretely emerging market. While in the beginning of the decade the AI-specific revenue accounted for a few millions, in the end of it, the market grew to a $2 billion industry (Russell & Norvig, 1995, p. 22ff).

Besides development in the research of robotics, computer vision, machine learning and representing complex knowledge-based systems, a holistic methodology and research agenda was strived for. To this day, the research activities' efforts in the field of artificial intelligence led to many successful implementations, augmenting human capabilities. Of those implementations would be traffic surveillance systems with automatic emergency calls in case of an accident, autonomous driving vehicles or knowledge-based medical systems, that can access large database and provide agreed upon evidences and scientific facts (ibid., p. 26 f.).

2.2 Types of Intelligence

Even though the concept of intelligence philosophically and psychologically has a different meaning, the entity can be abstracted and the (artificial) intelligence divided into different fields. Above all, this is reasonable in technical environments to precisely define the needed means for a methodological approach of problem solving. Moreover, the used mathematical models and methods differ for the specific use-cases.

On that account, neither the philosophical nor the psychological points of views will be discussed. The theoretical basics and mathematical methods would also go beyond the scope of this chapter, which is why they will not be discussed at this point.

2.2.1 Visual Intelligence

The term visual intelligence is understood to mean the ability to recognize and analyze patterns. In the human brain, this process runs subconsciously, with no possibility to activate or deactivate it by choice. On the contrary, visual intelligence of a machine can be determined by the detection of visually arranged impulses of information based on camera systems, infrared sensor or even digitally transmitted patterns in pictures. This kind of technology is used in iris and fingerprint scanner, face recognition or lane assist and machine vision systems for vehicles (Fig. 2.3).

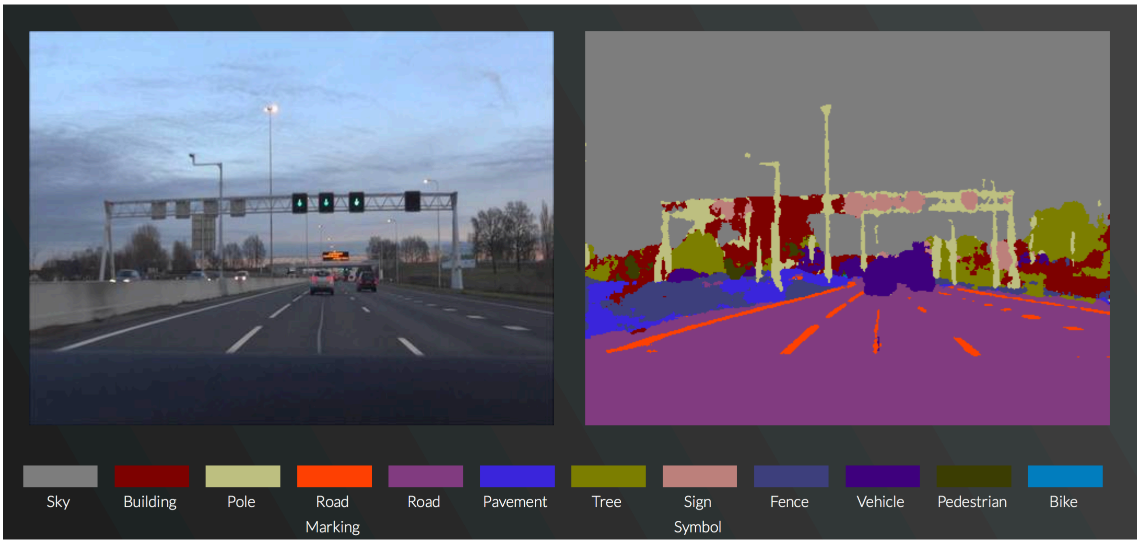

Fig. 2.3: Segmentation of live street view pictures (Kendall, et al., 2015)

If the image source is not used otherwise by humans, e.g. in manual surveillance mode, there is no reason in a) the display of the image and b) in the analysis of the original picture. In many situations, infrared cameras are used to better detect heat or patterns in gray scales then full color cameras (National Instruments, 2008). Nonetheless, there is a trend towards the complete analysis of real-time images, focusing machine learning applications.

2.2.2 Linguistic Intelligence

Linguistic intelligence represents the ability to convert speech to text and vice versa (Schroeder, et al., 2009). When addressing recognition by machines, there is a difference in speech recognition (linguistically) and text recognition (optical character recognition).

In the former case the analog speech signal first gets digitalized and compressed, before the signal is defined by three relevant parameters: time, frequency and intensity. Depending on the wording and phoneme, it will be determined which sound matches which word. Different mathematical methods will be applied, such as the fast fourier transformation (FFT), to analyze the signal (Fig. 2.4).

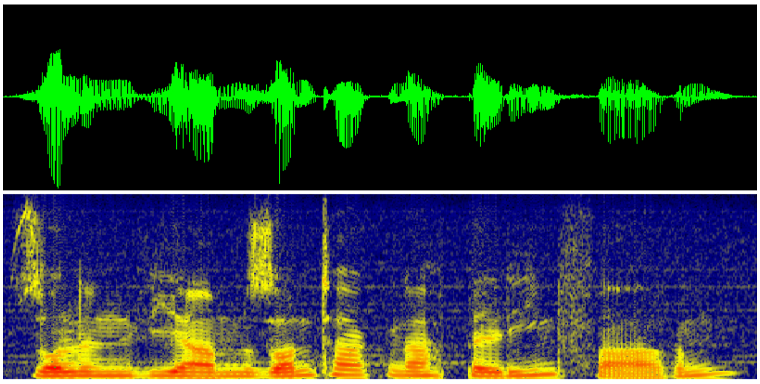

Fig. 2.4: Speech waveform and FFT-analysis of the sentence *Shall we drive to Berlin on sunday?* (Ger. *Sollen wir Sonntag nach Berlin fahren?*) (RWTH Aachen, kein Datum)

How the human brain masters speech recognition, cannot be describe with a mathematical model. Would speech recognition be based on words, humans could not recognize unknown words. Would it be based on phonemes, background noise or a different pronunciation would make the understanding of the wording impossible. The function of the speech recognition of the brain involves higher processing stages. This includes word recognition as well as syntactic and semantic analysis (Warth, 1999).

One step further lies speech understanding. This problem is solvable by machines, though it differs from speech recognition, for the meaning plays a necessary role. The machine is obliged to show a reaction based on speech. At this point, a cobot system, e.g. a cooperative man-robot system which is commonly used in production, can be referred to. Registering a verbal command, the cobot changes its initial state and fulfills the dictated task (Warth, 1999, cf. manipulative intelligence)

2.2.3 Manipulative Intelligence

Manipulating the environment, man and machine alike, for a specified reason comes under the term manipulative intelligence (Fromm, 2015). A robot does this manipulation as a one-time or constant action. Wielding and painting robots, assembly line robots or cobots as well as intelligent drones with special functions serve as examples (Schroeder, et al., 2009).

2.2.4 Rational Intelligence

Intelligence or knowledge which can be proven by facts, is called rational intelligence. The following of rules and drawing conclusions is part of rational intelligence. With regards to this kind of intelligence, machines most often work with databases and expert systems, so that specific information can be filtered from enormous databases for a specific task. In medical issues, an expert system can be searched for symptoms and give out relevant or linked information or even suggest treatment options based on factual experience (Schroeder, et al., 2009).

In recent times, this topic gets more relevant by the concept of *Big Data* and due to higher computing and storage capacities, faster data transmitting speeds and learning system algorithms. It is not seldom, that data protectionists criticize the issue of extensive data collections – in most cases consented for service value enrichment (Freitag, 2016). Big volumes of data, whether precious customer data to analyze buying behavior, machine data to optimize an assembly line or general data clouds that can now exist due to sinking storage prices, ought to be used by companies to maximize profits. The resulting analytical outcomes are based on rational intelligence. A human being would also be able to draw conclusions, in some cases maybe better owing to his intuition and experience, though, in consideration of the huge amount of data, the computing power of a machine outperforms any human being. Nonetheless, the evaluation of the visualized data is in most cases left for the human operator or data scientist.

2.3 Intelligent Entities – Man vs. Machine

The goal of developing artificial intelligence is the implementation of intelligent conduct in artificial entities – be it in the virtual or real world. This was, and still is, one of the fundamental objectives to exceed the human being in capacity and *brainpower*. Taken as a whole, this might be a long journey, but, as described, if the task is a calculable or logical sequence of steps, approximation and mathematical optimization, or decision trees and intelligent databases, i.e. tasks that depend on high calculating power, machines beat humans. When IBMs Deep Blue beat the then chess world champion Garry Kasparov in 1997, there was great astonishment. Not only was this the first win of a machine over a human, it was also a psychological milestone for the discipline of artificial intelligence. Chess, for centuries a benchmark for human intelligence and acumen, was now successfully challenged by a machine (Fine, 1997).

With regards to similarities to humans, a machine has an analogical but more abstract approach to processing information. An agent, an entity whether human or artificial that can sense its environment and respond to it, has to receive a stimulus that will be followed by a process and lead to an effect or the intention of it. For this, sensors and actuators are needed (Fig. 2.5). In a human, the sensors are his sensory organs, e.g. eyes, ears, etc., while the actuators are his muscle-controlled limbs, e.g. arms and legs. In a machine, cameras and ultrasonic sensors act as its sensors and engines as its actuators.

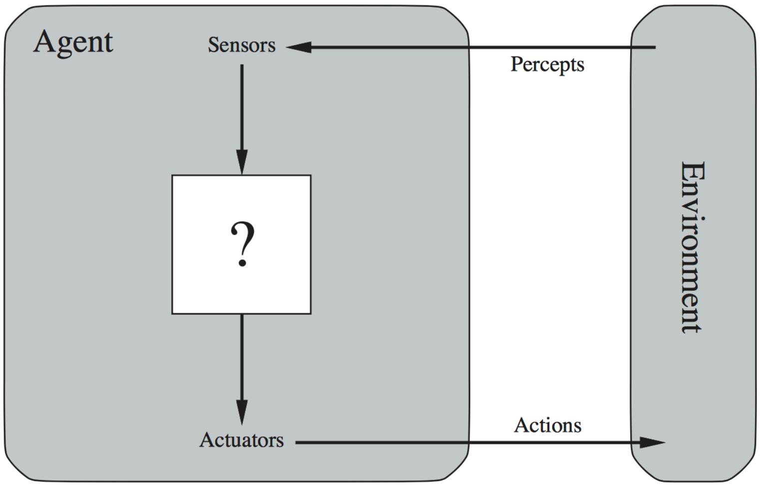

Fig. 2.5: Graphical representation of an agent (Russell & Norvig, 1995, p. 35)

In a software agent, data and digital values form the sensors and the display and routing of the evaluation the actuators. The above described concept however remains same regarding hardware as well as software (Russell & Norvig, 1995, p. 34).

The differences between human and artificial intelligence are numerous. Although in some tasks the machine outperforms humans, mostly because calculations in applications like autopilot or visualization of data clouds are more efficiently done by computers, humans are irreplaceable in areas where a value-based approach is necessary. The *human touch* will gain in importance in a world where machines fulfill tasks that were formerly done by humans (Lozovschi, 2017). *Artificial General Intelligence*, which refers to machine, that can perform a universal intellectual task that can originally only be done by a human, is a topic that lies in the future. In this light, there also exist no Turing machines, that can have a majority in their function (see subchapter 2.1). The closest to the Turing machine is *Eugene Goostman*, a chat bot program that could nevertheless convince 33 percent of skeptical testers to be a 13 years old boy from Ukraine in 2014 (Janschitz, 2014).

2.4 Benefits

The benefits and advantages of implementing artificial intelligence in existing and future systems are extremely promising.

Improvements in production, error accuracy, product development, computation and design, as well as navigating systems in unknown environments, and intelligent robots are just a small fraction of the possibilities that deal with addressing and implementing artificial intelligence.

With advances in artificial intelligence, new ways are found to improve products and increase efficiency: The antivirus software manufacturer Kaspersky relies on AI, for newly discovered malicious programs are 90-98% similar in code. By learning algorithms, these can be successfully detected (see spam filtering in chapter 3). In other areas, such as financial transactions in PayPal, an algorithm was successfully developed that could differentiate between transactions involved in money laundering and actual transactions based on legitimate business (Marr, 2016).

Many of these examples are based on immense amounts of data, which makes artificial intelligence a key technology of the coming decades. These large amounts of data are impossible for a person to analyze, let alone recognize patterns. Artificial intelligence is the lubrication of a technology gear that moves through many mechanisms. By further developing individual key technologies such as data transmission and infrastructure, system and processor architecture and the resulting digital business models, artificial intelligence can represent a unique interface between software and hardware.

2.5 Obstacles and Ethics

Nevertheless, these individual sub-areas of science also pose certain obstacles. Artificial intelligence cannot be fully exploited without the further development of new systems and infrastructure, such as the planned 5G network. For some applications, such as machine communication, perfect high-speed communication is mandatory. There is also talk of an end to Moore's law, the rule, which states that every 12 to 18 months the capacity of transistors doubles, after research projects at several Australian universities have successfully reached the physical limit of an atomic size for the development of a phosphorus-based transistor (Tally, 2012) Since the physical limit has been reached here, there is no way to ignore the fact that in other areas, such as quantum mechanics, there might be possible alternatives in order to meet the increasing demands for computing capacity. An undesirable consequence can be the increasing complexity due to competition, which means that new digital business models have to be introduced. This is a positive aspect, but it raises many people's concerns as to whether there is any added value in relation to the rising costs. One example is the connectivity of commercial aircrafts, which is still not economically achievable, since no one is willing to pay for this (still) expensive service (Bellamy, 2017). As a result, issues such as real-time data transmission, digital clones and autonomous aircraft are still far from being realized.

In addition to the technological and eco-political hurdles, social and ethical aspects also play a role. As early as 1942, the Three Laws of Robotics were formulated by Isaac Asimov, which state how the interaction between man and machine must be regulated in order to maintain harmony (Asimov, 1982, p. 67). However, the laws or rules that have to be followed only describe the behavior of the robot in order to prevent possible damage to humans. Originally, Asimov formulated the following three laws, which were slightly modified over time:

I. A robot may not injure a human being or, through inaction, allow a human being to come to harm.

II. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.

III. A robot must protect its own existence as long as such protection does not conflict with the First or Second Laws.

These laws state that the integrity of human health and life is the highest priority. It is also worth mentioning that the existence of a robot is subordinate to the human command. Accordingly, an order of automatic suicide would be legitimized by the second law. Law number three does not apply here, since in this case, there would be an evident conflict with law number two.

From an ethical point of view, there are several aspects that are thought to be relevant in near future, when artificial intelligence is applied to everyday use:

- Unemployment

  Speaking of automation, several physical labor types are in danger. Machines nowadays are being already used in fields like sorting, wielding, assembly, packaging, and many more. It is expected that 78% of predictable physical work, like assembly line work, and 25% of unpredictable physical work, like forestry or construction, can be replaced by adapting currently demonstrated technology (Chui, et al., 2016). Despite laborers losing their jobs, in several fields work fields an automation would result in fewer injuries, accidents, and fatalities, like in long-route freight forwarding or highly dangerous environments like wildfire incidents (Bossmann, 2016).

- Inequality

  Distribution of wealth is another point, that is on the agenda of the ethics of artificial intelligence. The economic system in today's world is based on compensation for contribution to the economy. In other words, laborers get paid for a service they offer or products they sell. With artificial intelligence on the rise, the distribution of wealth also experiences a shift. In 2014, the three biggest companies in Detroit and Silicon Valley had about the same revenue, only that in Silicon Valley the companies' employment rate were ten times lower than those in Detroit (ibid.) This results in a wider wealth gap in society.

- Humanity

Frequent interactions with machines are rising in numbers. From chat bots to service and product sales, many people start to interact with those machines like human beings. With those interactions, the machine gets to know more about the user and designs the user experience accordingly, speaking of clickbait headlines, which are often optimized with A/B split testing. Although this kind of software can be used to form society behavior more beneficial, the fear of using it greedily as a means to make money is present (ibid.).

- Bias

Machines more and more are designed to augment the work of humans, especially in fields where prediction of certain events are based on many parameters that cannot be done by humans. Since a machine is preset by humans with all the information it gets fed, there might be some sort of bias in some cases. Recent findings have shown that bias in the input data of a machine will affect the output. Especially if it acts in environments where there are human beings in play, for example when predicting future criminals, this is highly problematic (Bossmann, 2016). In these times of high technology acceptance, it is more important than ever to be aware of the consequences of a biased machine.

- Warfare and the security thereof

As more and more nations are building up artificial intelligence research programs, the military also takes part in developing new and intelligent systems for warfare. Unmanned drones are operated from a far distance, and equipped with machine vision, there would not even be the need for a human supervisor. As with every general technology, artificial intelligence might be used to kill other humans, which, from an ethical point, is viewed to be questionable. Moreover, where there are highly dangerous weapons, nations have to be aware of the higher security needs. Also, there is threat in the digital dimension, where cyber warfare has shown to be a serious danger (ibid.).

- Singularity

There are a lot of scenarios, from literature and motion-picture, about the downfall of humanity caused by intelligent machines. Formerly seen as a distant reality and even science fiction, calls are beginning to be made to regulate artificial intelligence and the use thereof. Humans are on top of the food chain due to their mental capabilities and intelligence. Hence it is concerning, that intelligent machines overcome humans in this regard and, what scientists define as *singularity*, replace the status of humans as most intelligent beings on this planet. Points are still open whether it is possible to regulate AI, or *pull the plug*, in case there is a threat expected (ibid.).

- Unintended consequences

Picking up the previous point, machines also can malfunction or cause threat in a non-intended way (ibid.). Giving capabilities to a machine that it cannot (, or was not programmed to,) take responsibilities for, may lead to unintended consequences. For instance, if the main task was to battle global warming and the machine comes up with a solution that is about shrinking the human population forcefully, the goal might be achieved, but not the way humans would have intended it.

- Rights of artificial intelligent entities

As there are rights of humans, and animals, there also might be questions about how to treat intelligent entities and robots. Questions like whether robots should be treated *humane*, or whether they should be rewarded or do something against their dislike, need to be answered. With machines getting more and more intelligent, also in their realistic interaction with humans, the feeling might emerge that robots can feel pain, too, thus affecting the way humans see and treat them. Time will show, how this issue will evolve; however, it is of importance to deal with the social status of robots and intelligent entities (Bossmann, 2016).

Especially in applications that concern society in general, these ethical aspects are being heavily discussed: Technological advances in autonomous driving, for example, have raised questions concerning human lives. An autonomous vehicle can also be called a robot in a broad sense. The question, how an autonomous car should behave in a situation where a threat to human life is inevitable, is subject of a social experiment at the MIT, which provides given scenarios for public consultation and discussion (Fig. 2.6).

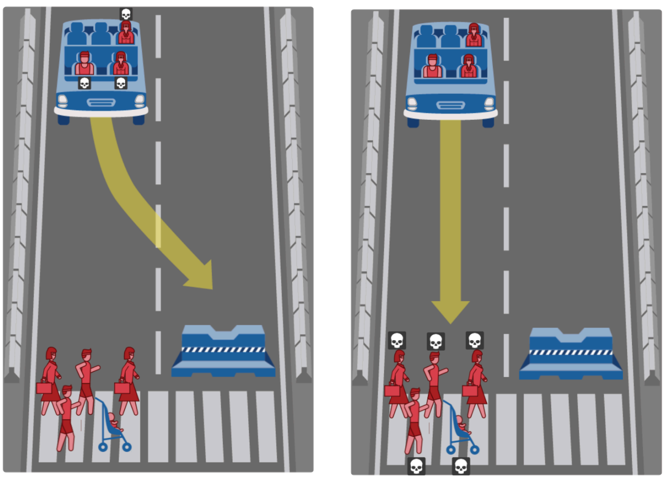

Fig. 2.6: Scenario of a moral dilemma of autonomous vehicles (MIT, kein Datum)

In above illustrated case, an autonomous vehicle drives on its lane when the car registers that there are pedestrians on its track and it cannot brake in time due to a technical defect. Either the vehicle changes its lane and crashes into a barrier, which would result in the death of all the passengers, or it remains on its track and thus inevitably kills the pedestrians. It is also possible that the losses may be weighed up against each other. How the importance should be weighted, whether a person's life (driver) is valued less than two lives (passengers), whether animals are valued less, or whether the life of a young person outweighs that of an older person in value or priority, are open questions. Finally, the question of liability arises as to who is responsible after such an accident and how the source of error can be located, or if it can be located at all. A code error is also imaginable, which will not be uncommon due to the large proportion of software in the vehicle and the fact that software programming gets outsourced to a large percentage in many cases (Reuters, 2007).

In even more blatant circumstances, some speak of concrete cases of a threat to mankind by machines. Recently, headlines about a Facebook chat bot appeared, which, according to various newspaper reports, has developed its own language for communicating with other bots, although this might be slightly dramatized to make a better story. Another fact is that this chat bot had learned to misrepresent information, e.g. lie, even though it was not explicitly programmed for this purpose (Maney, 2017). Achieving the goal of artificial intelligence of a program that can evolve, e.g. that a machine can program itself, can create danger for humanity. One such scenario is the encapsulation by developing an inaccessible new internet. At this level, regulation and control of the machines is unthinkable. Scientists and tech-entrepreneurs such as Stephen Hawking and Elon Musk warn against an apocalypse that could lead to an uncontrolled development of intelligent machines. Also, there is competition on an international level between nations to set up the AI industry. According to Musk, this is supposed to hold the potential of a third world war (Hern, 2017). Early regulation and a consciously created environment for further research are important for a sustainable and controlled introduction of highly intelligent entities.

2.6 Applications of Artificial Intelligence

The applications of artificial intelligence are vast in today’s society. Today, machines are designed for specific tasks, which make them per definition *weak AI* (in contrast to strong AI, or General Artificial Intelligence, e.g. a machine that can apply intelligence to a variety of problems rather than just one). However, in specific fields like speech recognition or games, AI is beginning to outperform humans. This chapter deals with the different fields artificial intelligence is applied to.

2.6.1 Computer Vision

One of the most researched topics in artificial intelligence is computer vision. Being an ability naturally existing in humans and animals, machines having vision is a challenge. Generally speaking, the goal of computer vision is to extract information and the transformation thereof through a visual input, most likely a single image or a sequence of images. Typically, the input data come from image sensors which the computing machine analyzes. Apart of analyzing and processing images, computer vision is also used in pattern recognition and guiding and controlling robots (BMVA, kein Datum). Applications of computer vision include the following examples:

- Automatic inspection (manufacturing)
- Controlling processes (industrial robot)
- Identification (quality inspections)
- Event detection (surveillance)
- Interaction with humans
- Object modeling (medical image analysis)
- Navigation (autonomous vehicle)

2.6.2 Natural Language Processing

Anthony Pesce defines natural language processing as *a field that covers computer understanding and manipulation of human language*. The main goal is to analyze, understand, and derive meaning from human language for further using. NLP is different to common word processing operations that view text as a string of characters in that it analyzes the hierarchical structure of a given text input: Several words make a phrase just like several phrases make a sentence. Above that, another goal is to create machines that can understand the meaning of human speech, which can be a tough task with the ambiguity of language (Kiser, 2016). Following points, among others, are examples of natural language processing:

- Chat bot
- Automated keyword generation
- Sentiment analysis
- Text summarization
- Extraction of topics from text
- Text mining
- Machine translation

2.6.3 Speech Recognition

In most cases, speech recognition refers to voice to text conversion. What sounds simple, are in reality complex-built architectures with huge vocabularies and grammar databases. Software built for speech recognition analyzes sound by filtering it into a readable format and making educated guesses what was said, or, what was intended to be said (Van der Velde, 2018). What started with machines that could understand numbers from zero to nine with a 90% accuracy in the 1950s, today's speech recognition tools like Google's Assistant or Apple's Siri reach higher accuracy than humans with an error rate under 5% (Boyd, 2018). Applications of speech recognition include following examples:

- In-car systems
- Language learning
- Speech-to-text for people with disabilities
- Automatic subtitling
- Court reporting
- Home automation
- Virtual assistants
- Robotics

2.6.4 Knowledge Representation and Reasoning

Knowledge representation and reasoning is about *symbolic encoding of propositions believed* (representation) and the *manipulation thereof to produce representations of new propositions* (reasoning). For example, a symbol like ♂ represents the male gender just like ♀ represents the female. The statement *John is Mary's father* indicates, that John is an adult male (Chaudhri, 2011). IBM's Watson for instance uses statistical and symbolic representation and reasoning to solve AI tasks. This way, Watson was able to beat humans in the *Jeopardy!* quiz show. Knowledge representation and reasoning is to be found in the following examples:

- Ontology databases
- Language understanding
- Expert system
- Scheduling and planning
- Resource allocation
- Model checking
- Games and puzzles

2.6.5 Other

Apart from the above explained applications, there are other specific fields AI being applied to solve problems and fulfill tasks. Of those are applications in robotics, games, as well as medicine and aviation, which will be addressed in the following chapters. Most of these fields are using the above explained goals, like machine vision, to give robots the ability to perceive their environment and interact accordingly. Although AI comes from computer science, it is not seldom that the applications take place in interdisciplinary work environments, for artificial intelligence is a general technology used in enhanced systems to solve tasks which usually require human intellect, and not just reserved for computer science. Most of these real-world implementations also use a variety of the above-mentioned applications. A broad field is automation of processes. Those processes can be from every industry, like improving and accelerating aircraft maintenance jobs in aviation, observation of laboratory work and quality assuring in medicine, or others, using several AI techniques. An autonomous vehicle for instance, can use all of the four mentioned applications like machine vision, natural language processing etc. to interact with its environment.

3 Machine Learning

This chapter deals with the basic principles and methods of machine learning, and specifically learning algorithms. Learning algorithms are characteristic for machine learning in that sense, that the algorithm is able to learn from data. Learning considered by itself is defined by Mitchell as *learning from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E* (Goodfellow, et al., 2016). In simpler words, it means learning is by doing action and measuring *if* the outcome improves with the learnt. Machine learning helps solving tasks, that are too difficult to solve with prewritten and fixed programs (Goodfellow, et al., 2016, p. 99).

An overview of possible types of tasks (T) is given in Table 3.1 according to Goodfellow, et al. (2016, p. 100 ff.).

Table 3.1: Overview of some types of tasks solvable with machine learning algorithms

Type of task (T)	**Explanation**	**Example**
Classification	Matching an input to certain given categories	Face recognition and automated tagging of people (Facebook); Reverse image search (Google); Robot waiter that recognizes food and delivers them on guest's command; Detecting and classifying ice on an aircraft's skin
Classification with missing inputs	Probability distribution over all the relevant variables to solve classification tasks when an input is missing	Specific medical tests that are either expensive or not possible to carry out; Faulty flight data with missing input values
Regression	Prediction of a numerical value based on an input	Pricing of a house based on data; Prediction of expected flight cancellation insurance claim; Algorithmic stock trading
Transcription	Observation of unstructured data representation and transcription	Extract address numbers into text form (Google Streetview);

	into discrete, textual form	Transcribe audio wave form into text (Microsoft, IBM, Google, Apple);
Machine translation	Conversion of a sequence of characters into the same data format	Language translators in general (Google, DeepL); Decoding of a PIREP (pilot report) into readable text
Structured output	Any task where the output is a vector (data structure containing multiple values) with important relationship between the different elements	Mapping of a natural language sentence into a tree to describe its grammatical structure (parsing); Pixel-wise segmentation of aerial photos to determine roads in a map; Guiding a pilotless aircraft through the airport runway through cameras
Anomaly detection	Sifting through a data set flagging unusual or atypical objects or events	Credit card fraud and misuse detection through customer's purchasing habit; General time series data with a connected value
Synthesis and sampling	Generation of new examples that are similar to the training data	Generation of texture picture files for video games; Machine reading of written text for the blind; Generation of aircraft cabin interior training data in cases of data shortage for health monitoring of cabin furniture
Imputation of missing values	Predicting the values of missing entries of a given new example	Probability model of flight overbooking given extensive private information of passengers (demography, earnings, social media activity)
Denoising	Prediction of clean example from its corrupted version	Digitalization of poor quality scan of written reports or books;
Density estimation / probability mass	Explicitly capture probability distribution of data structures (the	Use cases of imputation of missing values (see above) due to known data distribution through density estimation

function estimation	methods before were designed to implicitly capture those data distributions)	

Just like the tasks differ from case to case, the measurable performance (P) is specific to that task. In cases of classification, classification with missing input, and transcription, the accuracy of the model is often measured, e.g. how many correct outputs were recorded. Alternatively, the error rate, e.g. the percentage of wrong outputs, can be measured. In other tasks like density estimation, a continuous-valued score is given for every example, e.g. the farther a data point is from the density center, the lower the score. However, every task itself requires a performance measure that corresponds well to the desired behavior of the system, which is not always an easy job (ibid., p. 101 f.). The fine-tuning of the performance measure is up to the system designer.

Experience (E) in this context would mean that the learning machine has experienced a training data set and is now able to perform the algorithm accurately on new, unseen examples.

3.1 Learning Machines

Machine learning is about exploring and finding computer algorithms to perform defined tasks, as described above. The artificial generation of knowledge from experience is also referred to as machine learning. If a system receives enough examples, it recognizes patterns and can generalize them. A central point here, however, is that the machine itself draws conclusions from the findings which were not previously programmed this way. The main focus lies therefore on the automated learning process. It should be learning to complete a task without the user or programmer assisting. An example of filtering spam mails is given below. Instead of writing code to filter spam from the mailbox, algorithms are being developed to allow the machine to write its own program to filter spam messages (Schapire, 2008, p. 1).

One way of setting up a rule would be to analyze spam mails beforehand. An analysis shows that 80% of spam mails contain the word cheap (Fig. 3.1).

Fig. 3.1: Representation of rule-based spam filtering (Serrano, 2016a)

From this fact that 80% of the e-mails containing the word cheap, a rule can be derived. However, this also includes a smaller part of the non-spam mails, which can be prevented by further rules. Additional characteristics of spam mails can be that they often contain spelling mistakes or no subject line (Fig. 3.2). Based on these findings, a set of rules is being established according to which the program knows how to categorize emails (Serrano, 2016a).

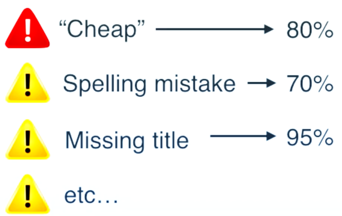

Fig. 3.2: Probabilities for suspicious keywords in spam emails (Serrano, 2016a)

The next step for the program would be to analyze the emails marked as spam and recognize patterns. In this way, the database of possible spam keywords can be expanded and compared with the clean emails. Manual entries, such as manual marking of spam mails, are also a possibility for the machine to learn further. If the spam attacks should change due to changed internet surfing behavior or more aggressive spam tactics, it is literally pre-programmed that the machine learns and adapts from them.

This concept of an ever-learning machine is a central element of artificial intelligence. Musk, Tesla's CEO, announced that test tracks totaling 10 billion kilometers will have to be driven before the worldwide legal approval of autonomous vehicles is granted. At the end of 2016, Tesla's autonomous column would complete 5 million kilometers of valuable learning material for AI every day. Toyota confirmed this number with its own estimate of 14.2 billion kilometers, which is slightly above Tesla's target (Ohnsman, 2016).

And this is exactly how there are many other areas of application for machine learning: recognition and digitization of handwriting, face recognition, generation of topic headings for scanned newspaper articles, speech recognition, medical diagnoses, customer segmentation, fraud prevention, as well as weather forecasts (Schapire, 2008, p. 1f).

There are different models for machine learning, which are mainly differentiated into parametric (logistic regression, naive Bayes classifier, artificial neural networks, etc.) and non-parametric algorithms (K-Nearest-Neighbor, decision trees, etc.). These differ in the fact that parametric algorithms require fixed parameters to process a data set. This approach is particularly suitable for tasks in which the user must have a certain amount of prior knowledge, e.g. a defined data set. Non-parametric algorithms, being much slower, are suitable for tasks where there is no prior knowledge or no defined data. The next subchapter introduces the method of artificial neural networks, which plays an important role in the field of deep learning.

3.2 Methods of Machine Learning

Machine learning methods are various in numbers. According to the kind of experience the learning algorithms are allowed to have, one way to categorize machine learning algorithms is into *supervised* and *unsupervised* algorithms. Besides, there are other methods, that can be used supervised as well as unsupervised, like artificial neural networks.

With supervised learning algorithms, each example of a dataset usually is associated with a label (Fig. 3.3).

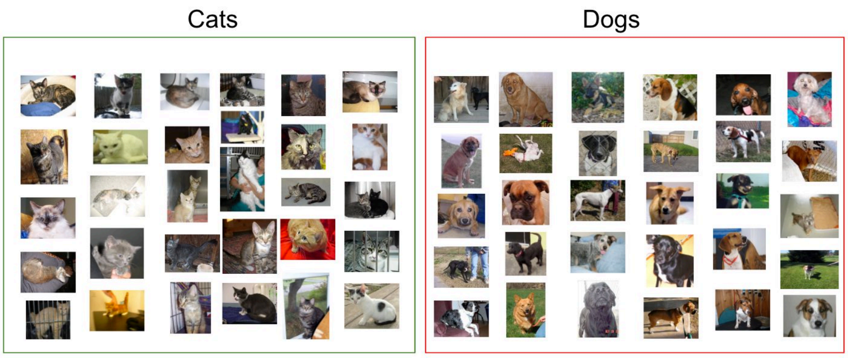

Fig. 3.3: Samples of labelled cat's and dog's pictures from the Kaggle dataset (Moujahid, 2016)

The term supervised also comes from the view of an instructor or teacher, who determines what the picture, or in general the data point, should be viewed as.

An unsupervised learning algorithm on the other hand has to make sense of the data without an instructor or teacher. Usually an unsupervised algorithm is used for learning the probability distribution of a dataset or clustering, e.g. dividing a dataset into clusters of similar examples.

Another form of learning algorithm is *reinforcement learning*, where the machine does not experience a fixed dataset. A reinforcement learning algorithm interacts with the environment, so that there is a feedback loop between the learning system and its experience (Goodfellow, et al., 2016, p. 103 ff.). Lately, reinforcement learning has made major headlines with DeepMind's AlphaGo and its ability to learn *from scratch* (Fig. A.1). By playing against itself, and given rules of the game, the machine could learn much faster than previous deep reinforcement learning algorithms (Hassabis & Silver, 2017). In the following subchapters, some usual supervised as well as unsupervised learning algorithms are presented in detail. The concept of artificial neural networks are presented in detail in subchapter 3.2.3.

3.2.1 Supervised Learning Algorithms

As explained before, one of the characteristics of a supervised learning algorithm is the match-making of inputs x and outputs y. In some cases, the data cannot be gathered automatically and needs a human supervisor; either way the term *supervised* still applies. This subchapter deals with a variety of supervised learning algorithms that are used in machine learning applications.

3.2.1.1 Logistic Regression

The machine learning method of logistic regression is based on estimating a probability distribution and used in classification use-cases. While linear regression handles two possible classes (0 and 1; and therefore, only one classes' probability has to be known to determine the probability of the other, for both must sum up to 1), logistic regression has to deal with several value classes. In subchapter 2.4.1 the probability model was classified with a straight line, which is characteristic for linear regression, based on the probability of error of each data point (see Fig. 2.12 and 2.13). In general terms, a linear regression model is used to predict the value of a certain point, logistic regression is used to predict binary values such as the probability of getting diabetes based on factors like age, weight, gender, etc. (Griffith, 2017). The above figure represents such a logistic regression model, where the probability of passing an exam versus the hours spent studying is shown (Fig. 3.4).

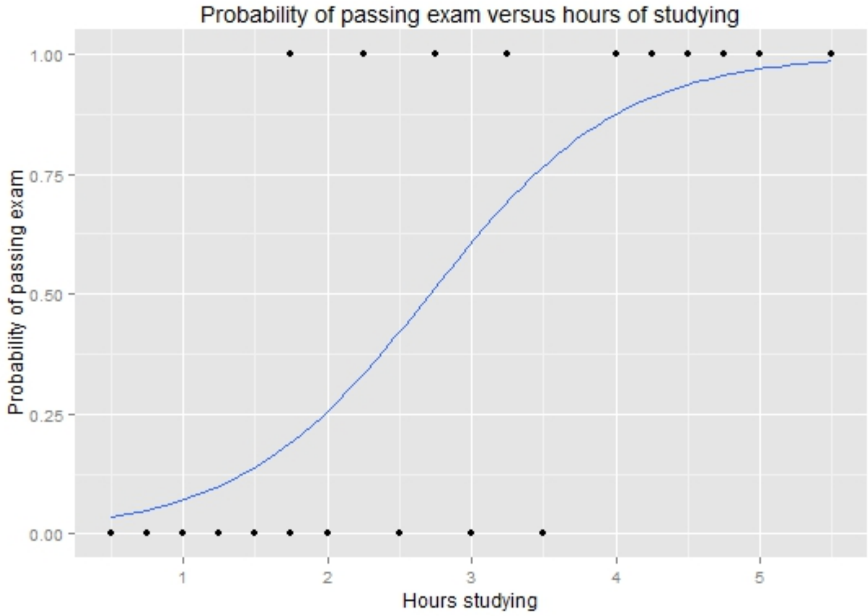

Fig. 3.4: Basic logistic regression to model the probability of passing an exam versus hours of studying (Griffith, 2017)

However, it is said by Goodfellow, et al., that a linear regression model can be portrayed by solving the equations, whereas a logistic regression needs a numerical optimization method called *gradient descent* (2016, p. 140). To better understand machine learning theory and basics, the definition of gradient descent is given as followed: The issue of gradient descent is the mathematical optimization problem of

$$\min_{x \in \mathbb{R}^n} f(x),$$

where based from a starting point, the steepest descent is chosen, until the lowest possible point is reached. In most cases, the direction of descent points towards the minimizing of errors. The figure below shows a visualization of the gradient descent optimization problem (Fig. 3.5).

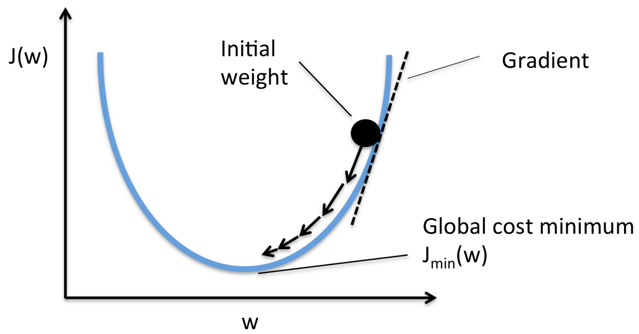

Fig. 3.5: Illustration of the gradient descent concept with $J(w)$ as the cost function and the initial weight seeking the lowest point (Raschka, 2018)

3.2.1.2 Support Vector Machines

Another strong classifying technique is the Support Vector Machine (SVM), which differs from the logistic regression model in one respect, that it does not provide probabilities but only class outputs. Simply put, an SVM searches for the best possible separation of data points. In the view of a two-dimensional split, best possible means the widest gap between two classes of data (Fig. 3.6).

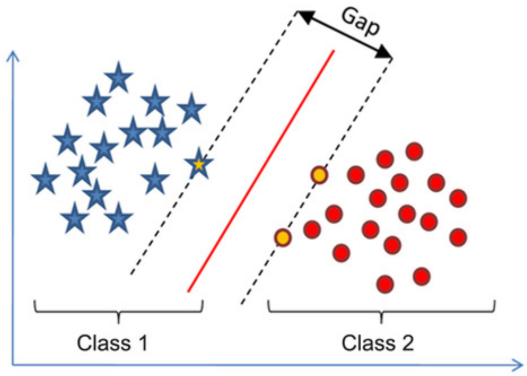

Fig. 3.6: Visual representation of two data classes, separated with the widest possible gap; the yellow highlighted data points portray the support vectors (Sharma, 2015)

Where SVM's especially come into play in an elegant way, are data point distributions that are not separable with just one line. Goodfellow, et al., describe the *kernel trick* as a key innovation that SVM's brought into play efficiently as a means to split complex data constellations (2016, p. 141 ff.). To be more precise, Fig. 3.7 is given as an example below.

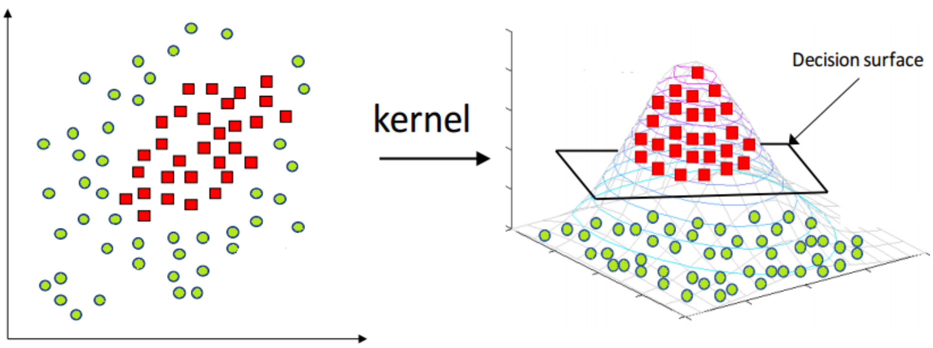

Fig. 3.7: Computing a non-linearly separable function into a higher dimension linearly separable function using the kernel trick (Jain, 2017)

In this case, two classes of data are distributed in a way that a line cannot separate. To solve this problem, a kernel function

$$K(\vec{x},\vec{y}) = e^{-\frac{\|\vec{x}-\vec{y}\|^2}{2\sigma^2}} \tag{1}$$

which can be seen as a three-dimensional landscape function of x as the vector in the shown plane and y as the distance to the center of the landscape. σ marks the ring size of the kernel. With this method, it is possible to split data with a linear function which makes handling complex data easier, even though this is done by a linear plane. There are several kernel functions; the one shown above is known as the radial basis function (RBF) kernel.

3.2.1.3 k-Nearest Neighbor

The k-nearest neighbor algorithm is one of the simpler yet effective algorithms. One of the key characteristics is, that there is no training stage. A new output can be generated for a given test input by running the algorithm on the input data point to class it to the correct data group (Goodfellow, et al., 2016, p. 143 f.). k in this regard means the number of observing neighbors to the test data input. In Fig. 3.8, an example of a new test input (star) is shown with two possible outputs, depending on the number of neighbors (3 and 6, respectively).

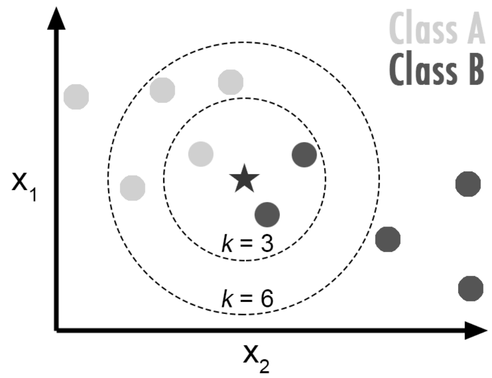

Fig. 3.8: Classifying a new data point (star) with the kNN algorithm (Ivan, 2017)

In the case of the figure shown above, a new data point (start) is registered and needs to be classified. The k-NN algorithm makes predictions using the dataset directly. For this, the entire dataset is searched for the k most similar instances (neighbors) and the distance to all of the neighbors is being calculated to determine to which class the new data point is more similar. k ranges from 3 to 10 for most datasets (Ivan, 2017).

3.2.1.4 Decision Trees

Decision tree algorithms are used both for classification (categorical output, e.g. yes / no) and regression (continuous output, e.g. a number like 34) of a test dataset. What is given, are observed rules that (should) apply to all of the gathered data. New data points are then being classified, moving forward through the nodes step by step. To make it easier to understand, Fig. 3.9 is given below.

Decision trees are used when there are binary values to attributes of a dataset, like if the wind is strong or weak; a continuous value is also possible. In case of the below figure, if the person should play golf or not, the question is being asked, what the weather outlook looks like. If it is cloudy, the answer is directly *Yes*. If there is rain forecasted, the question applies how strong or weak the wind is, to get a definite answer. The same goes for the humidity levels, if the weather is sunny, with a high humidity level resulting in a *No*, and a normal level in a *Yes* to playing golf.

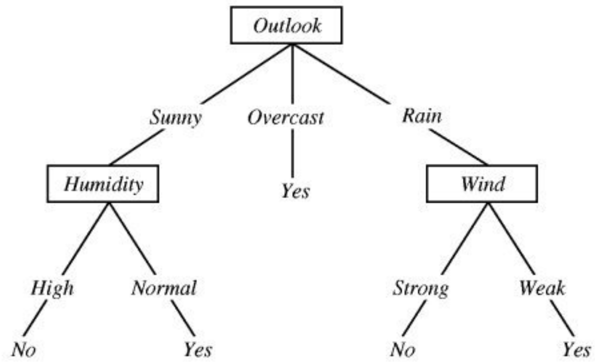

Fig. 3.9: Decision tree with a discreet output value to play golf or not (HV, 2017)

Classifying the data and applying a decision tree-based set of rules only works when the learning dataset is observed beforehand. For example, if there is a dataset of various data points given with respect to two variables X1 and X2, the first task is to define data regions (Fig. 3.10).

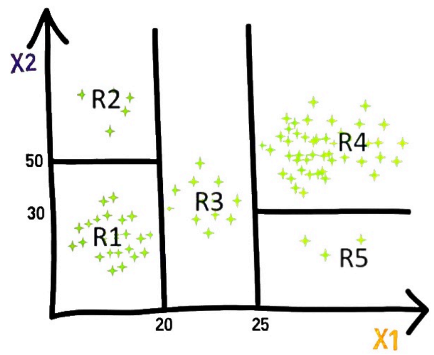

Fig. 3.10: Defining five data regions of the observed dataset (Singh, 2017)

After defining the data regions, there are several choices on where to begin with the decision tree. One choice would be to first ask if the value of the new data point is smaller than 20 in regard to the first variable X1, and if so, whether the second variable X2 is lower or higher than 50, to classify the new data point as R1 or R2, respectively. The figure below shows a complete decision tree to this solve this example (Fig. 3.11).

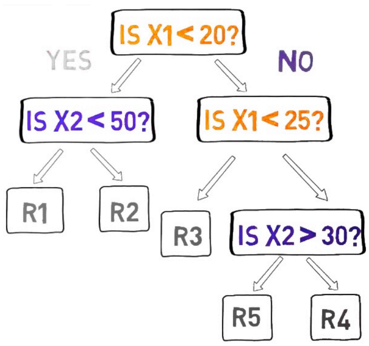

Fig. 3.11: Decision tree in accordance to the distributed data points in Fig. 3.8 (Singh, 2017)

Despite being simple and effective in nature, the prediction time and the time invested in building the algorithm are relatively high (Ivan, 2017).

3.2.2 Unsupervised Learning Algorithms

The main difference between supervised and unsupervised learning algorithms is that with an unsupervised task, there is only input data, but no corresponding output variables (Brownlee, 2016). Unsupervised learning machines extract information from a data distribution that do not require human labor to define examples. Mostly, these techniques are used when the task is to cluster examples into groups of examples, density estimation or learning to draw samples from a distribution. One typical task would be for an unsupervised learning machine to find the best representation of the data. This might not work for every kind of task since *best* can be highly subjective, however generally speaking, it defines a representation that preservers as much information as possible while maintaining penalties and constraints (Goodfellow, et al., 2016, p. 146 f.).

3.2.2.1 Principal Components Analysis

The principal components analysis algorithm is an algorithm to learn represent data constellations. Large sets of data can be reduced with this method in size and number of dimensions, hence also used in data compression. It is also a means to identify patterns in data, simplifying and highlighting the similarities and differences of the data in itself (Smith, 2002, p. 12). One example of a direction aligning task of a dataset is shown in the figure below (Fig. 3.12).

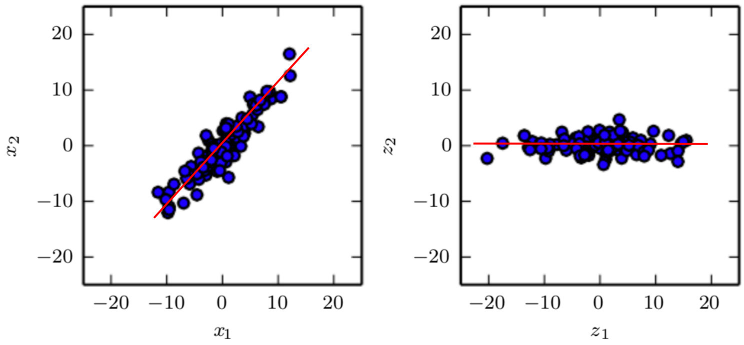

Fig. 3.12: Linear projection and alignment of a data cloud along the main axes (Goodfellow, et al., 2016, p. 148)

In terms of image recognition, this algorithm works just well with, for example, pictures of faces, that are aligned in a row after the PCA identifies statistical patterns in the image data regarding differences and similarities. The new image will get aligned at its right place in between the processed dataset (Smith, 2002, p. 21 f.).

3.2.2.2 k-means Clustering

k-means clustering is defined as an unsupervised algorithm that divides the dataset into k different clusters that are near each other (Goodfellow, et al., 2016, p. 150). Before looking at the data and labelling it, this method allows to find clusters in a dataset that have formed organically. Thus, finding the groups of data makes assumptions possible whether and how new data points are labelled due to the pre-clustering of known data (Trevino, 2016). In the figure below, the driving behavior of 4,000 drivers are shown with the distance feature on the x-axis and the speeding feature on the y-axis (Fig. 3.13).

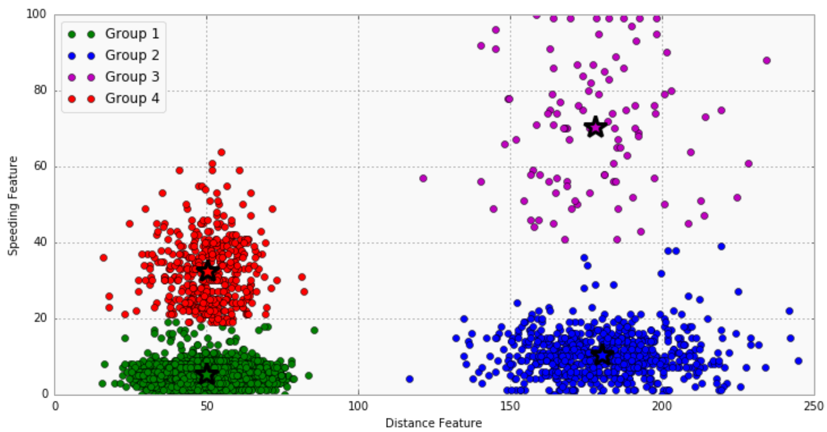

Fig. 3.13: Clustering of drivers' data based on two features: distance and speeding (Trevino, 2016)

This algorithm can be used in different business application, for example:

- Detecting activity types in motion sensors
- Identifying groups in health monitoring
- Categorize inventory by manufacturing metrics
- Behavioral segmentation based on activity on platform or application

and more (Trevino, 2016).

3.2.3 Artificial Neural Networks

The concept of artificial neural networks goes back to McCulloch and Pitts in the early 1940s (see Chapter 2.1). The topic of neural networks has recently gained enormous importance. Not only has the effectiveness of this method increased as a result of increasing computing power, but topics such as big data and new sensor technologies have also increased the efficiency of the output of results. Another factor are increasingly complex problems and patterns addressed by findings in the field of artificial intelligence, thus increasing the efficiency of this method. As described in section 2.4, fixed parameters are required for this method in order to obtain a desired output.

The concept is based on the model of the neuron in the human brain (Fig. 3.14).

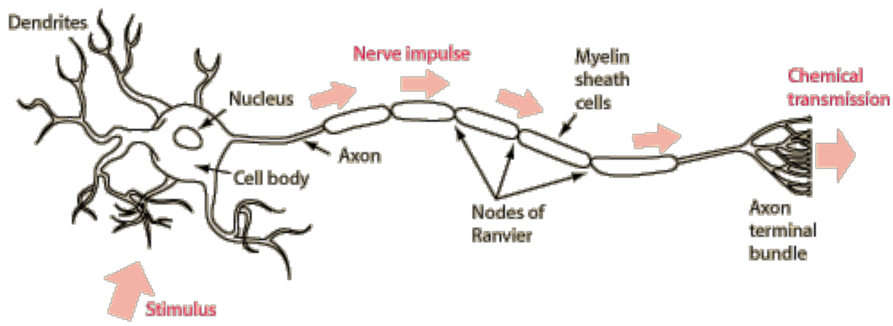

Fig. 3.14: Model representation of a neuron (Guthix, 2014)

In the case of the neuron, the cell receives electrical input signals through its dendrites via synapses. If this signal is sufficiently large enough, an electrical impulse is given out by the axon to the synapses that are further connected. If the pulses are not strong enough, no impulse is transmitted. Researchers of artificial intelligence also use this model to create a simplified mathematical model, which functions similar to the nerve cell (Fig. 3.15).

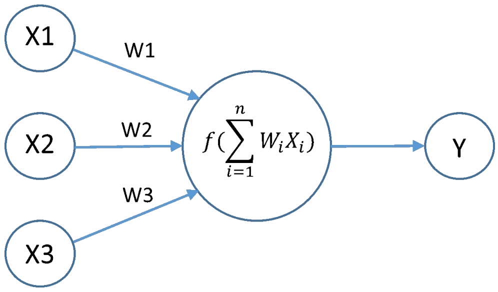

Fig. 3.15: Model of an artificial neuron (Anomaly, 2016)

In the case above, three inputs X1, X2 and X3 (dendrites) with specific weights (W1, W2, W3) go into the neuron (nerve cell). A signal moving forward through a neuron is determined by the incoming weighted sum of inputs and the activation function (see subchapter 2.4). Applying the activation function on the input sum (and the set bias of the neuron) either make the neuron *fire* the information out (and how intense), or not, depending on the function's outcome (Moujahid, 2016). Three common activation functions are given below (Fig. 3.16). Different tasks require different activation functions.

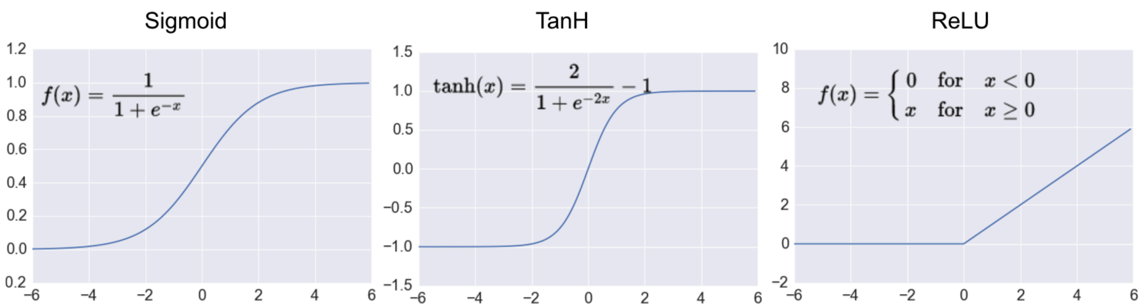

Fig. 3.16: Three different activation functions; Rectified Linear Unit (ReLU) is the most common one (Moujahid, 2016)

The learning process takes place when the desired output is compared with the actual output. If there is an error, the weights will be adjusted accordingly and again tested. This procedure is called backpropagation (Mankar, 2011). Following figure shows the control loop of this process (Fig. 3.17).

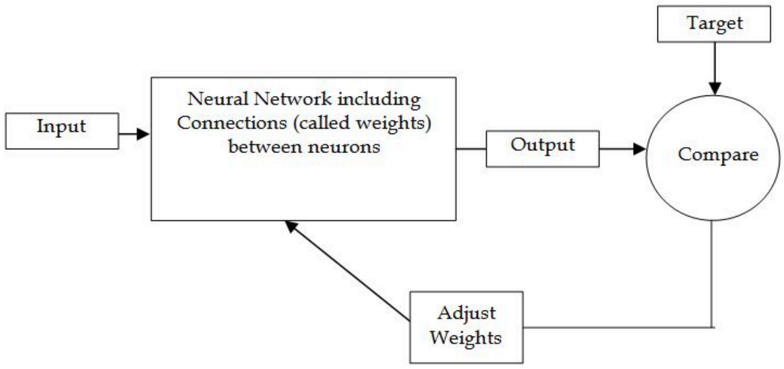

Fig. 3.17: Control loop of the backpropagation process (Mankar, 2011)

In the case of a simple but complete model of a neural network as shown below (Fig. 3.18), there are three inputs, a hidden layer with four neurons and one output.

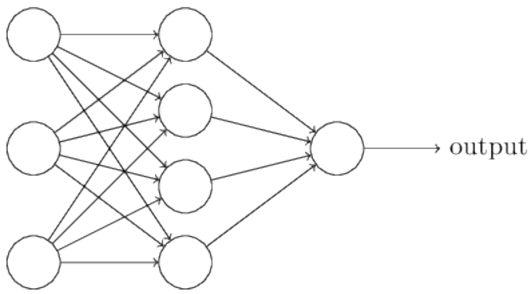

Fig. 3.18: Model of an artificial neural network of three layers (Nielsen, 2017)

The next figure below shows a part of a neural network that is designed to recognize handwritten numbers (Fig. 3.19). The 3 input nodes are different inputs, here three randomly selected digits 0, 5 and 9. This procedure is called classification. The hidden layer is about recognizing the 0: In order to successfully recognize and confirm that the scanned number is a 0, the digit is quartered, and each corner is compared individually. The weightings of the paths are initially distributed equally. During the first run, a result is randomly displayed which is not definite since the paths are equally weighted. Based on this first assumption, the path weights are adjusted and continuously improved according to the trial and error principle (Nielsen, 2017).

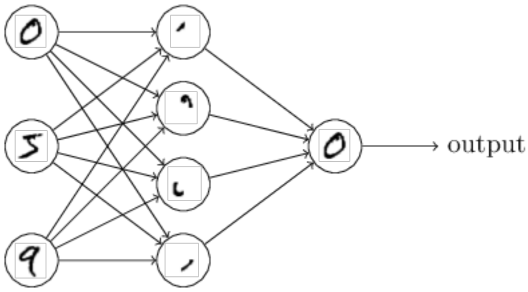

Fig. 3.19: Artificial neural network for character recognition (according to Nielsen, 2017)

The threshold values of each neuron can also be changed, but in this case, this would not be necessary. The change in the weights is based on an error delta, also called cost or loss, which is calculated after each run. At the lowest possible cost value, which is calculated from the sum of all products of weight and threshold, the result is acceptable. Since the result (output) is already known, the weights of the paths from the 0 input to the hidden layer are now multiplied, as well as the weights from the 5 to the first neuron and those of the 9 to the fourth neuron are increased slightly, since the shapes in these two cases are relatively similar. Generally speaking, knowledge is always stored in the weights (Beck, kein Datum).

Classification of data is a central component of machine learning applications. Another generally applicable example would be the logistic regression, a method for modelling and analyzing discrete variables. Classification may also be carried out, which represents a further step. A coordinate system with several data points of different types can be used for this (Fig. 3.20).

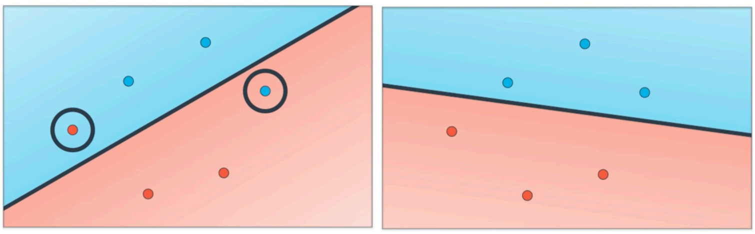

Fig. 3.20: Target task: Separating data points with a straight line; left two errors (circled), right none (Serrano, 2016b)

In this case, a random straight line is placed through the data points and two errors are registered. Typically, the errors are quantified. A sum of the error sizes, i.e. how far they are from the separation line and whether they are on the correct side, is formed with the aim of varying the straight-line function so that the error is as small as possible. In this case, the requirement of the smallest defect size is fulfilled.

However, if this example is complicated by far more complex and quantitatively increased data points, this task that cannot be solved with a straight line. Here, an artificial neural network can be one solution (Fig. 3.21).

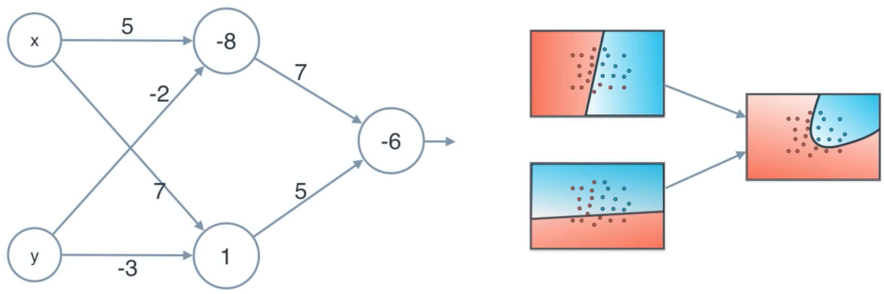

Fig. 3.21: Integration of several regression analyses into an artificial neural network (Serrano, 2016b)

When looking at the individual neurons, it becomes clear that in the hidden layer, there are the individual functional lines that intersect the dataset. Since this is not possible with a single straight line, a weighted distribution results in a quadratic function that represents an image of the two straight lines.

The linear equations in this case are

$$y_1 = \frac{5}{2}x - 4$$

and

$$y_2 = \frac{7}{3}x - \frac{1}{3}$$

If these are converted, they result in

$$5x - 2y = -8$$

and

$$7x - 3y = 1$$

so that the individual variables and constants are implemented into the neural network.

With the search for the lowest error ratio, the weights can also be varied and improved. Specifically, in the case described above, the first incoming neural activation is weighted seven-fold, while the second one is weighted five-fold. This is how a higher-order polynomial placed in the data field that separates the two data types from each other. The -6 in the last neuron is a randomly selected number and has no further effect.

This way, much more complex data clouds can be separated by increasing either the number of neurons in the hidden layer or the number of hidden layers themselves. It can be assumed that the more complex the task and the richer the range of information, the deeper the neural network goes with several hidden layers. This is then called a *deep* neural network.

To address the *learning* in a neural network, gradient descent and backpropagation comes into play. In a simple supervised example, when training the neural network, the output is a wanted goal to achieve. Since at first the weights of the neurons are distributed randomly, by changing those weights, a smaller error is aimed for. The procedure is done from output layer backwards until the input layer since the slope of each layer depends on the layer before. This process is called backpropagation (Geng & Shannon, 2017). To summarize the learning concept, it can be said that it goes forward in the network through the layers to compute the loss, then backwards through the entire layers to compute the gradient and adjust the weights and biases (Li, et al., 2017).

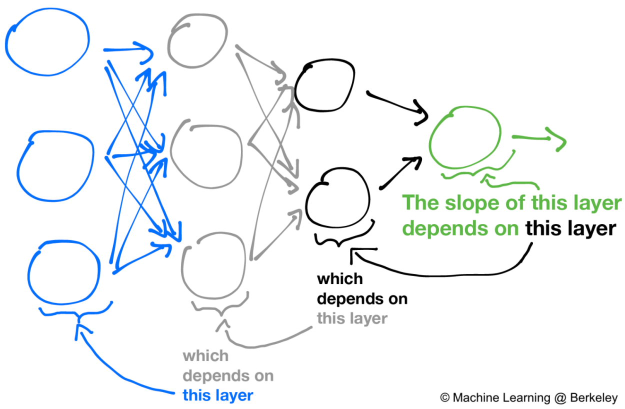

Fig. 3.22: Concept of backpropagation from output to input layer (Geng & Shannon, 2017)

Above figure explains the concept of backpropagation further (Fig. 3.22).

Although backpropagation is a doable task for simple neural networks, when it comes to deep neural networks (see Chapter 4), the time for training takes much longer. The cause for this is a phenomenon called the vanishing gradient. This is mainly the result of the activation function. Sigmoid and tanh activation functions squash the input into a very small output range, and additionally, in a non-linear way. The result is, that a large change in the input results in a smaller change in the output, which makes it worse when more layers are stacked up (Garg, 2015). A solution would be to take another activation function, e.g. ReLU, or use other models like Restricted Boltzmann Machines or Deep Belief Nets (Hinton, et al., 2006; Bengio & LeCun, 2007). A further theoretical explanation of these models would be out of the scope of this thesis.

3.3 Limitations of Traditional Machine Learning

The machine learning algorithms described in the previous subchapters all work very well on various real-world problems concerning handling and analyzing data. However, everything that goes beyond simple and processed data, like image or speech recognition or high-dimensional data, the traditional approaches tend to be insufficient. With more complex data, the computational cost also rises, which makes it impossible to work on new problems with traditional techniques. Deep learning was developed to solve these and other issues (Goodfellow, et al., 2016, p. 155).

With traditional machine learning, to be able to classify whether a cat is shown in the picture or some other animal, features need to be predefined. Possible features might be to prove if the animal has whiskers, if it has ears and if so, then if they are pointed. With deep learning, this procedure is on another level: the algorithm finds out the features, which are most important to classify the pictures, where in machine learning, all features had to be preset manually.

With deep learning, in general, edges that are most relevant are identified first. Based on the edges, combination of shapes and edges are looked for. Several more stages prove, if these edges and shapes match to a bigger picture, and if so, the algorithm decides on basis of activation and probability what the output should be.

Deep learning is a subset of machine learning, and often the terms are being used synonymously, however, there are crucial differences. The main difference lies in the feature engineering. As shown in below figure, extracting the features is an additional step to the deep learning algorithm (Fig. 3.23). It is not just difficult but also time-consuming, and it relies on the domain expertise of the supervisor. With deep learning algorithms, a more end-to-end approach is promised (Moujahid, 2016).

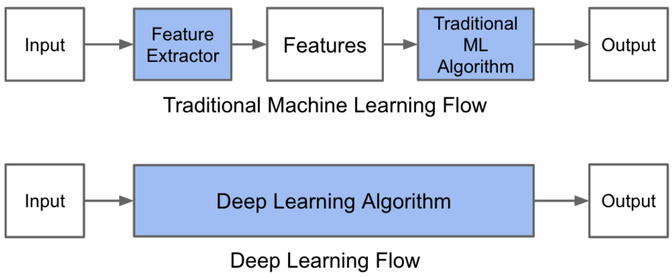

Fig. 3.23: Comparing the work flow of traditional machine learning to deep learning (Moujahid, 2016)

More differences are given in the table below, as cited by Shaikh (2017).

Table 3.2: Comparison of machine learning and deep learning regarding image recognition tasks (Shaikh, 2017)

	Machine Learning	**Deep Learning**
Data dependencies	Runs well with small amount of data; Scalability limits regarding to efficiency	Need large amount of data; Performance increases with increasing amount of data
Hardware dependencies	Low-end hardware requirements	High-end hardware requirements (GPU)
Feature engineering	Features (pixel values, shape, position etc.) need to be identified and hand-coded by expert	Deep learning algorithm *hierarchically* learns high-value features directly from data
Problem solving approach	Breaking down of the problem into different parts (object detection, then object recognition)	End-to-end processing
Execution time	Short training and execution time (depending on algorithm seconds to hours)	Long training time (up to weeks) and very short execution time
Interpretability	Easy to interpret algorithm reasoning / outcome	Difficult to impossible to interpret the outcome (though mathematically possible to determine why some nodes were activated, and some not); Biggest obstacle for industry use is the lack of interpretability

4 Deep Learning

Deep learning is based on neural networks with a minimum of two hidden layers, thus it is called *deep*. Andrew Ng described deep learning in 2013 as an attempt to use learning algorithms based on brain simulation and make progress in machine learning (Ng, 2013). This vague formulation became more explicit in 2015, when Ng compared deep learning to a rocket that needed both an oversized engine and a huge amount of fuel to fly. This was made possible by breakthroughs in other areas and technological developments, such as the possibility of cloud computing or high-performance computers - as an analogy to rocket propulsion - and the huge amounts of data as fuel, generated by the internet and micro sensors that are becoming smaller and smaller. The decisive advantage over other learning algorithms is that in deep learning the performance curve also increases with increasing data volume, whereas older learning algorithms achieved an unsurpassable performance plateau. Deep learning achieves an unmatched efficiency in fields like speech recognition, image recognition and detection, forecasting, and natural language processing over traditional machine learning approaches (Ng, 2015). Based on a talk by Ng (2016), the following representation was built to further ease the understanding of the advantages of deep neural networks (Fig. 4.1).

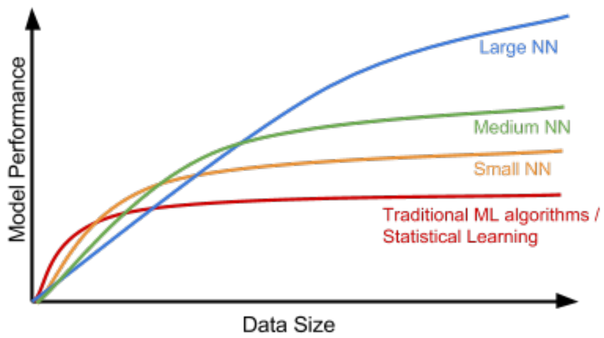

Fig. 4.1: Qualitative representation of model performance over data size (Weng, 2017)

Another reason why deep learning achieved this massive attention is the flaws of classical machine learning algorithms that were limited in their ability to process data in their natural and raw form. Building a pattern recognition system would require lots of engineering hours and a lot of expertise to design a feature extractor to transform the raw data, e.g. pixel values of an image, into a suitable representation or feature vector from which the learning model could detect or classify patterns in the input (LeCun, et al., 2015).

The next subchapters will give a deeper insight into the architecture of deep artificial neural networks and the use thereof.

4.1 Advanced Concepts of Neural Networks

As already described the basis of artificial neural networks in subchapter 3.2.3, and how it aims to process information similar to the human brain, this chapter gives a short insight about neural network models used in deep learning algorithms. In subchapter 3.2.3 several examples of feedforward neural networks (perceptron; see next subchapter) were shown (see Fig. 3.13, 3.16 ff.). Besides this, what makes deep learning so powerful, are the advanced concepts of neural networks, that drove the accuracy up and made it more applicable in domains like computer vision, pattern recognition and forecasting. The concepts of multilayer perceptrons (MLP), convolutional neural networks (CNN), recurrent neural networks (RNN), general adversarial networks (GAN) and reinforcement learning will be presented in this chapter. These types of neural networks are commonly used in deep learning applications. Since there are limited patents to deep learning models, various modified versions are found for different tasks. Additionally, in scientific journals every new or modified version is compared to known models in terms of accuracy and efficiency to point out superiority of their specific method and model development (Redmon, et al., 2015; He, et al., 2018). This competition drives the development of new concepts.

4.1.1 Multilayer Perceptron

Since single neurons are restricted to linear calculations and not able to solve complex tasks, there is the need for multilayered neural networks (Riedmiller, 2010). As already shown in Fig. 3.18, 3.19 and 3.22, the multilayer perceptron is a feed forward neural network with a minimum of one hidden layer. The concept is to have neurons work as logical operators, as already stated in detail in chapter 3.2.3. Having more than one hidden layer makes it a *deep* neural network, as shown in the figures below (Fig. 4.2, Fig 4.3).

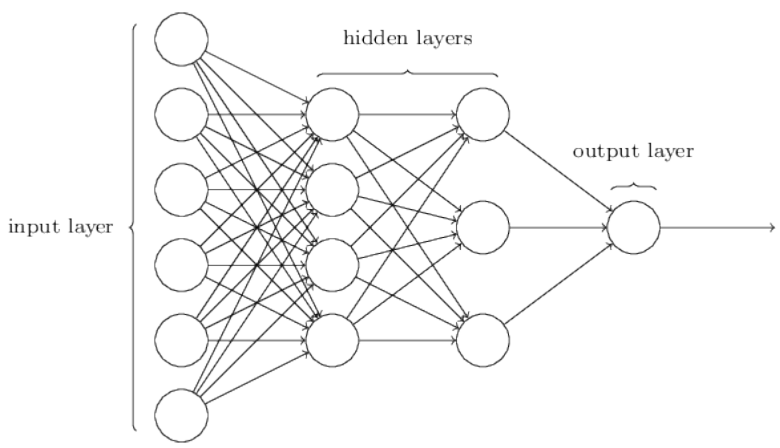

Fig. 4.2. Model of a perceptron with two hidden layers (Karim, 2016)

Multilayer perceptrons can be used in many applications from handwritten character recognition and pattern classification to prediction and approximation. However, deep neural networks have radically improved the efficiency of classifying images and data in comparison with other methods of machine learning. A specific use case was given in the introduction to Chapter 4 and subchapter 3.2.3. Following figure shows a pixel by pixel-based approach (28 by 28-pixel images) of a multilayer perceptron for detecting handwritten digits, which contains 784 = 28 x 28 neurons in the input layer (Fig. 4.3). The figure is only showing a reduced input layer since 784 input neurons would be hard to represent in this medium. Also, there is no need to show the whole number of neurons to get additional information out of said figure.

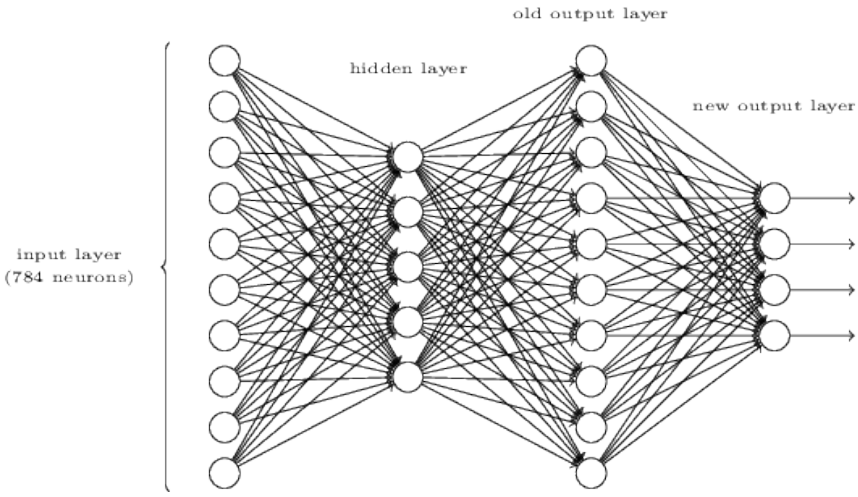

Fig. 4.3: Multilayer perceptron for recognizing handwritten digits (Nielsen, 2017)

4.1.2 Convolutional Neural Network

Convolutional neural networks are similar to the ordinary (deep) neural network (multilayer perceptron) from the previous subchapter. The main difference, beside the architecture, is the explicit assumption that the inputs are images. With the concept of local receptive fields (see below) and parameter sharing, the number of parameters in the network are reduced (Karpathy, 2018). Just on a side note: If a regular neural network would be used for an image of 200 by 200 pixels with three color channels, it would result into neurons that have 200 x 200 x 3 = 120,000 weights.

To better understand the concept of the convolutional architecture, below figures are shown (Figs. 4.4 ff.).

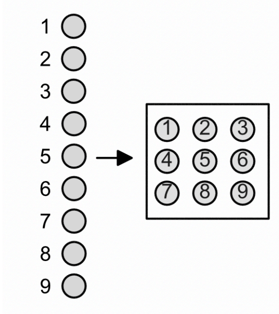

Fig. 4.4: Rearrangement of the neurons into two dimensions (Magruder, 2018a)

The one-dimensional neurons are rearranged into an array of two-dimensional neurons. Thinking of an image, two dimensions make sense, when the concept of local receptive fields is explained further: Local receptive fields represent a way of extracting local features (and later combining them) based on the work of Hubel and Wiesel on neurophysiology of how humans perceive information through several layers of *filters* (Hubel & Wiesel, 1962). To understand the concept better, below shown figure is presented (Fig. 4.5).

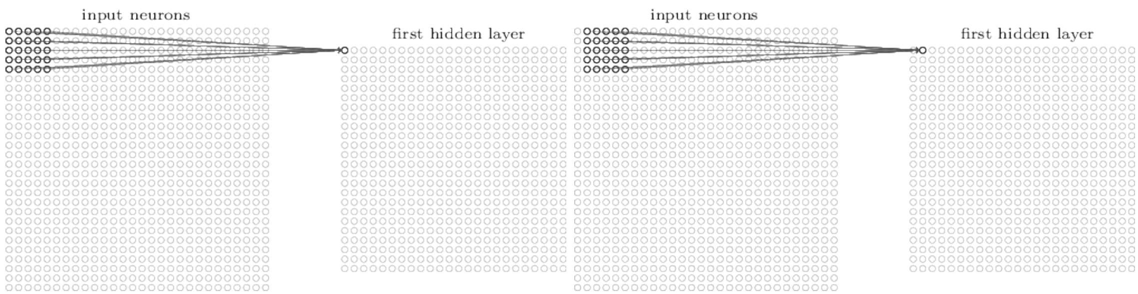

Fig. 4.5: Concept of local receptive fields to detect features in every part of the image (Nielsen, 2017)

The defined region of the input layer, in the above figure it is 5 by 5 neurons, gets connected to one neuron in the hidden layer. For a 28 by 28 neuron *map*, there are 24 by 24 unique local receptive fields. Unlike regular neural networks (multilayer perceptron), the weight and bias of the hidden layer neurons are all the same for that particular hidden layer. The term for this is called shared weights and shared bias respectively. That means that throughout the first layer to the hidden layer, also called feature map, the same feature is being detected over and over again. ReLU is most commonly used as the activation function to get rid of negative entries in the neurons of the hidden layer. To detect other characteristics, there can exist several feature maps (Nielsen, 2017). Moreover, the system will gradually improve its performance by adjusting the preset random weights and biases with the help of gradient descent (Ruder, 2017; Serrano, 2017a).

After the just described convolutional layers, there comes a pooling layer as represented below (Fig. 4.6).

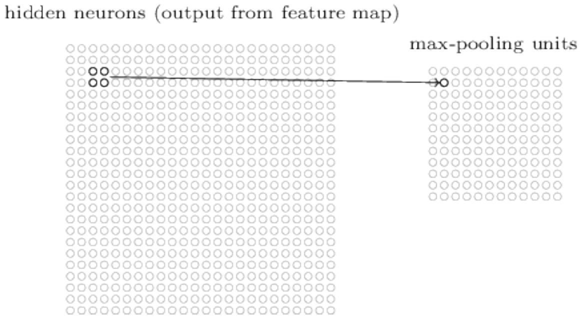

Fig. 4.6: Pooling layer, condensing the feature map (Nielsen, 2017)

Pooling describes the process of preparing a condensed feature map by taking the maximum output of a 2 by 2 region of neurons, as an example linked to above figure, and connecting it to the specified neuron of the pooling layer. Taking the maximum output is also known as max-pooling (ibid.).

The detection of features continues until a prediction based on the low to high level features can be made, which is the case when connecting the last hidden layer to one or more output layers. Before the fully connected output layer, there can be several convolutional layers with feature maps and pooling layers, as shown in the figures below (Fig.4.7 and 4.8 respectively).

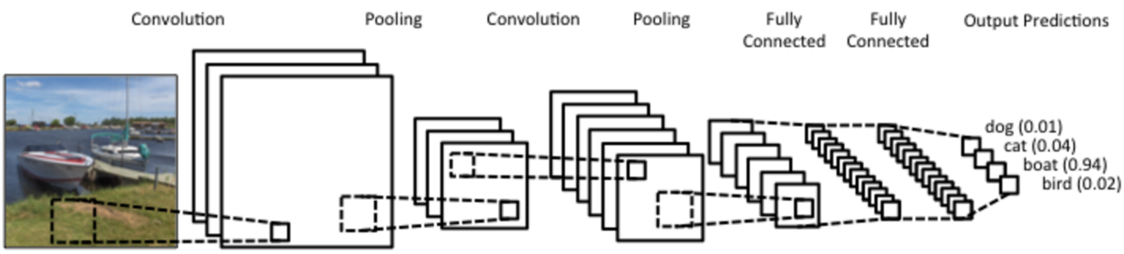

Fig. 4.7: CNN with two convolutional layers to predict boat images (Long, 2017)

As described above, there are several levels of feature recognition, for which an example is being represented by below figure.

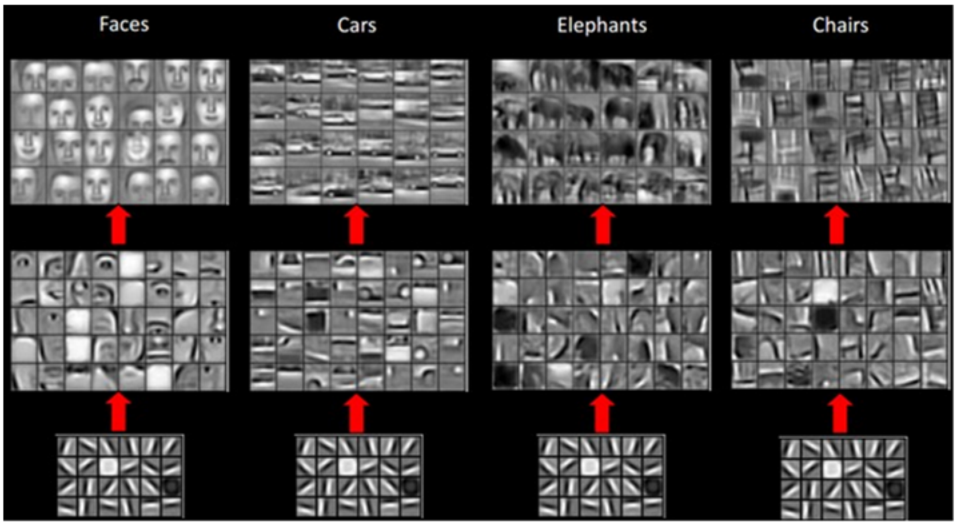

Fig. 4.8: Four examples of different feature maps from low to high level features (Dernoncourt, 2015)

The above-mentioned procedure is as of now the state-of-the-art in object recognition problems. Applications like self-driving cars and machine vision all were heavily influenced by the development of convolutional neural networks. On top of the classic CNN architecture, several other models were built, including R-CNN, Fast R-CNN, Mask R-CNN, and YOLO. While in 2012 Krizhevsky, et al., shook the computer vision world with the only deep learning-based image recognition model with an accuracy of 85 percent (the second-place algorithm was only performing with a 74 percent accuracy), CNN-based image recognition outperformed humans with 95 percent in 2015 (Krizhevsky, et al., 2012; Mallick, 2016). CNN's are widely used in the industry by Google, Facebook, and Apple (LeCun, 2016; Koehrsen, 2017; Parziale, 2016).

4.1.3 Recurrent Neural Network

Characteristic for the neural network models described before (multilayer perceptron, convolutional neural network) were that they are feedforward networks, i.e. the input determines the activation of all the neurons throughout the remaining layers. The model is static and outputs with the same quality. In a recurrent neural network however, there is a notion of dynamic change over time; this is especially helpful in analyzing data or processes that change over time. The behavior of hidden neurons can be determined not only by the activation in previous layers, but also by the activation of itself at an earlier time step, as described below. This is especially useful in natural language or speech translation (Nielsen, 2017).

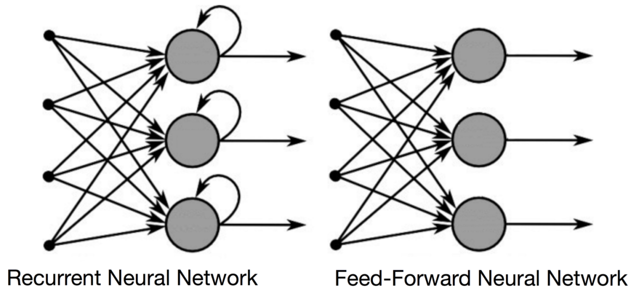

Fig. 4.9: Representation of the information flow in a recurrent neural network (Dongens, 2018)

As seen in above figure, the neuron takes into account what comes as regular input as well as what it already has processed from the time step before. This might be useful for processes that change over time. A very simple example would be a recommendation system for a canteen based on different inputs such as weather and season, costs for groceries, available skills (of employees, e.g. for exotic food), forecasted utilization, and previous meals cooked (Serrano, 2017b). Other examples are given below.

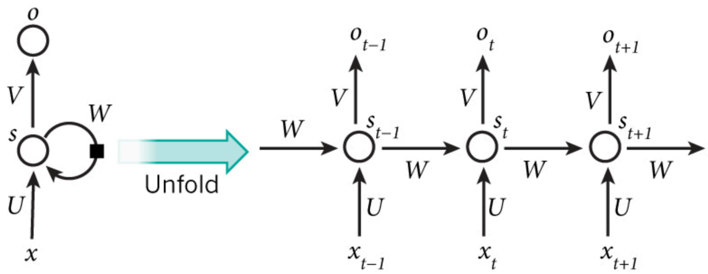

Fig. 4.10: Unfolded RNN; each time step has the current state, input, and output (Britz, 2015)

As shown in above figure (*x* representing the input, *o* representing the output, and *s* the state, as well as the parameters *U*, *V*, and *W*), an unfolded RNN represents each state (in other words memory) at a given time step, with the time step before as an additional input to build a time-series of interrelated states and outputs. In the example of the canteen, this would mean x inputs of environmental constraints (e.g. weather and season, costs for groceries, available skills, forecasted utilization) to process the ideal main dish (output) for the given date (time), with the additional constraint (W) that the main dish shall not be the same as yesterdays (or last 14 days).

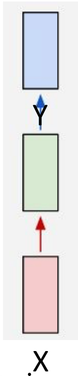

Fig. 4.11: Vanilla RNN; X (red) represents input, Y (blue) represents output, hidden neuron (green) represents the core of the RNN (see Li, et al., 2017)

To focus more on the process sequence, a different representation for RNN's will be chosen for the rest of this subchapter (Fig. 4.11 ff.). The term *vanilla* in this context refers to the basic model of a neural network, or single hidden layer backpropagation network (Hastie, et al., 2008, p. 392). In the following figure, different process sequences are shown. The main differentiation lies in the qualitative number of inputs and outputs (Fig. 4.12).

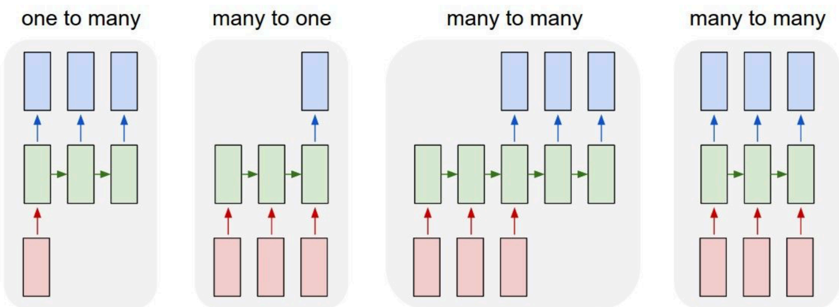

Fig. 4.12: Four different cases of conceptual RNN models (Li, et al., 2017)

In the case of *one to many*, an image may be used as an input, processing it through the RNN to give out a sequence of words, for example to caption an image (see subchapter 5.1.4 for a specific use case). One can think of a CNN, which was presented in the previous subchapter, as an encoder that extracts information out of the image, connected to an RNN that decodes the information into language (Radhakrishnan, 2017a). A simple example is shown below (Fig. 4.13).

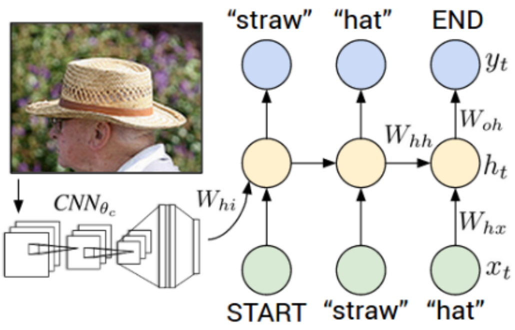

Fig. 4.13: Image captioning of a *straw hat* (Radhakrishnan, 2017a; Radhakrishnan, 2017b)

Referring to the figure above, the first word output *straw* is being put as the input in the next time step, so that, according to the probability of the next word, *hat* is being put out after another run of the RNN. The output of the last step, in this case *hat*, being the input of the next, gives out nothing that could *probably* be the next word of the sentence. Another example would be Google's recommendation for a search query when typing half a sentence or question in to the search bar (Fig. 4.14).

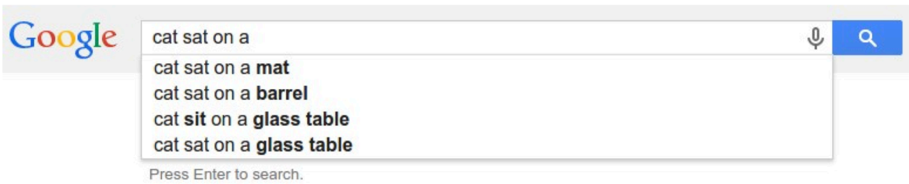

Fig. 4.14: Automated completion of unfinished sentences in Google's search engine (Karpathy, 2015)

Many to one on the other hand would make sense in sentiment analysis and classification for example, where the input are many words and the output the matching sentiment. *Many to many* however, comes into use in machine translation, e.g. translation of a sequence of words from one language to another, or classification of a video on a frame by frame level (Li, et al., 2017).

The main difference between recurrent neural networks and feedforward neural networks is the learning method. Since backpropagation (see subchapter 3.2.3) is a gradient based method, the problem of the vanishing gradient applies to recurrent neural networks, too. Additionally, each time step is the equivalent of a full layer in a feedforward network, e.g. training a 100 time-step RNN would take as long as training 100 layers of a feedforward network. This leads to exponentially small gradients and a decay of information through time (Rajagopal, 2015). Below figure represents the situation of the vanishing gradient and the decay of information in recurrent neural networks (Fig. 4.15).

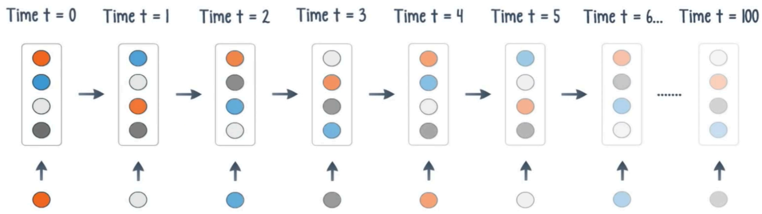

Fig. 4.15: Decay of information over time in a recurrent neural network (Rajagopal, 2015)

To get a better sense of the problem: For a sentence like *I grew up in France… I speak fluent…* the word *French* is most probably the best choice, since the RNN has learnt the context of the sentence by the point it was all about France. Imagining a longer gap, it gets harder for the RNN to forecast the next word, e.g. in longer sentences, where key elements are split apart through other sentences. Fortunately, there are several ways to address this problem, of those are gradient clipping, better optimizers, steeper gates, and gating. The latter is one of the more popular ones, more specifically Long Short-Term Memory (LSTM), which helps the networks to decide when to forget the current input and when to remember it for future time steps (Rajagopal, 2015).

Long short-term memory is the result of Hochreiter & Schmidhuber's research in 1997, which defines a memory cell using logistic and linear units with multiplicative interactions (Hinton, et al., 2016). Below figure represents the concept visually (Fig. 4.16).

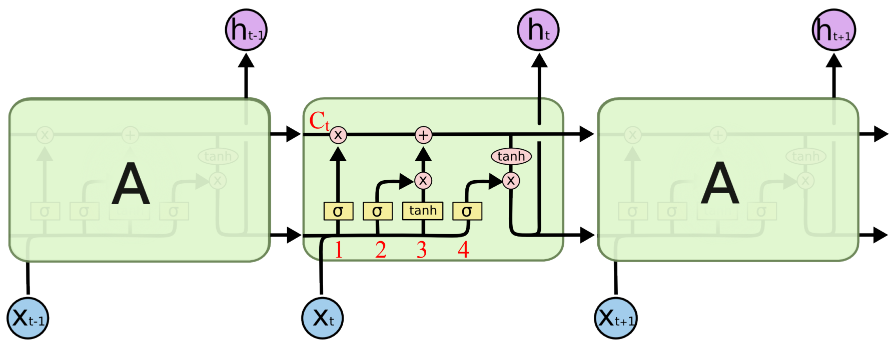

Fig. 4.16: Representation of an LSTM cell with four interacting layers (yellow) and pointwise operations (red) (Olah, 2015)

In the first step, in the *forget gate layer*, it is being decided which information to keep and which to throw away (1). The information goes through the layer and a sigmoid activation function, on basis of the previous and current input either marks it with 1 (information goes through) or 0 (information does not go through, i.e. forgetting the information). The next layer is about which information to let through and store in the cell state C_t. The *input gate layer* decides which values get updated (2) whereas the tanh layer creates a vector of new candidate values that could be added to the state (3). In the context of a language model, this might be the gender of a new subject, which replaces the old one.

Before the *output gate layer* (4), the cell state needs to be updated. This is done by getting the old cell state, forgetting the inputs that were decided to forget in the first layer and giving out new candidate values from the second and third layer. To decide what cell state to give out to the next cell as its input, a sigmoid layer comes into play. The tanh function then maps the values between -1 and 1 and multiplies it by the output of the sigmoid layer. In the example of the language model, the LSTM cell might put out the information whether the subject is singular or plural to that the verb (which might probably come next) gets conjugated correctly (Olah, 2015).

4.1.4 Generative Adversarial Network

The algorithms presented before were all of discriminative nature, e.g. classifying input data. Discriminative algorithms map features to labels, giving out probabilities of certain inputs being of which class. In contrast, instead of predicting a label on basis of the given features, generative algorithms predict features on basis of given labels. In the spam filtering task, the question a generative algorithm would rather be *Assuming this email is spam, how likely are these features?* (Gibson, et al., kein Datum) One of the newer concepts in deep learning using this approach is the generative adversarial network, short GAN. This relatively new framework was proposed by Goodfellow, et al., for estimating generative models. Two models are trained simultaneously: A generative model that captures the data distribution, and a discriminative model that estimates the probability of an input being from the training dataset rather than from the generative model. Goodfellow, et al., themselves give an analogous example of the generative model being a team of counterfeiters, producing fake currency and using it without detention, whereas the discriminative model is analogous to the police, trying to detect the fake currency. The competition drives both teams to improve their methods until a unique solution of a 50 percent chance of detecting the fake is the result (Goodfellow, et al., 2014). This adversarial process is just like the situation in a minimax game in game theory (Ramachandra, 2017).

Below figure represents the above explained concept in a visual and sequential manner (Fig. 4.17).

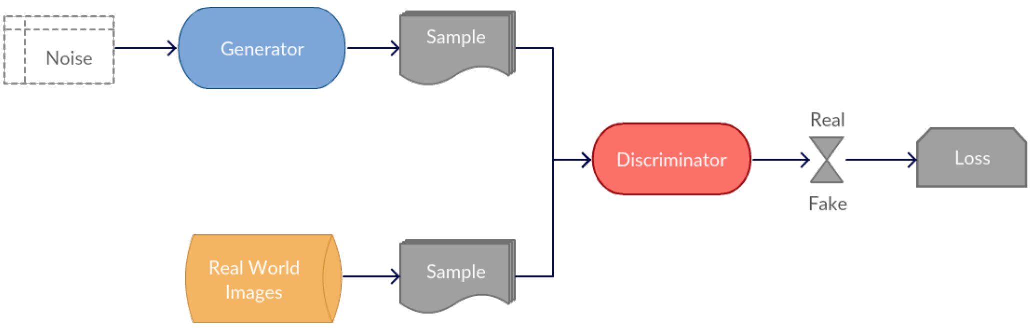

Fig. 4.17: Concept of GANs shown in a sequential representation (Fortuner, 2018)

A prior noise is going into the generator, putting out a sample of the processed input which goes into the discriminator. Likewise, the sample of a real image also goes into the discriminator. The discriminator then decides with a sigmoid function if the inputs are real or fake. In the beginning, the loss of the generated image is high, resulting in a higher gradient. As the net is trained, the images are getting more and more similar to the real-world images, thus the loss shrinks. A well-trained state is achieved, when the discriminator cannot classify the fakes as fake (Shibuya, 2017).

GANs are applied most straightforward when both models, generative and discriminative, are multilayer perceptrons, as proposed by Goodfellow, et al. (2014). However, GANs (so called Creative Adversarial Networks as proposed by the paper) were also applied to CNNs to generatively produce art (Elgammal, et al., 2017).

Immense potential is being offered with generative adversarial networks. Of the many use cases derived from this technique, there are models like the conditional generative adversarial network (cGAN) that were built in recent years and that can give out photo-realistic results (Isola, et al., 2017). Of those examples, few are shown below. (Fig. 4.18).

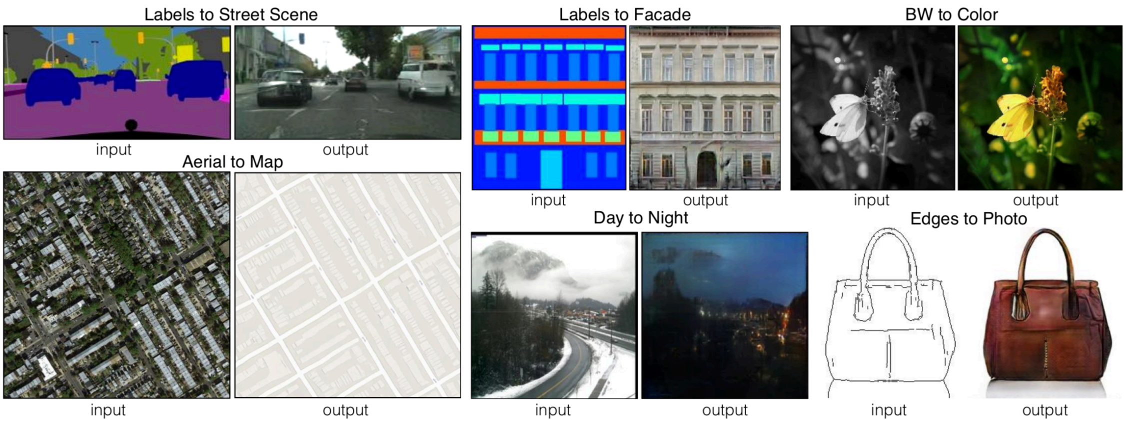

Fig. 4.18: Examples of image-to-image translation tasks with cGANs (Isola, et al., 2017)

Image-to-Image translation is a promising application. Just like a letter can be in a specific language, images can be in a specific setting, too. From gray scale images, filtered channels to drafts and sketches, there are possibilities to translate those kinds of images with the help of GANs. In the paper in which the above represented results were published, street maps could be derived from aerial images (Isola, et al., 2017). In the context of this thesis, this can be interesting in generating airport maps. Since streets and similar terrains like airplane bases get changed over time, airports or aviation companies can therefore generate their own maps and reduce costs for satellite imagery. In a broader view on possibilities in the industry 4.0 and with business models based on individualization of products, interesting applications can be molded with the power of GANs. With the help of easier production methods like 3D-printing, simple graphical user interfaces (GUI) are thinkable where the user gives a simple input like a sketch and the AI translates this to an actual product.

Because of the possibility to recreate new data from a given dataset, GANs are thought to be one of the most interesting and promising ideas of the last twenty years of deep learning research (Castelvecchi, 2017).

4.1.5 Reinforcement Learning

Recalling the definition of an agent from subchapter 2.3, reinforcement learning is based on fulfilling tasks by trial-and-error, striving for rewards and evading punishment (Silver, 2016). Reinforcement learning agents therefore learn to maximize their expected future rewards from interaction with an environment, as Sutton & Barto defined.

A Markov decision process is a framework designed for stochastic environments to help make decisions. The goal is to find a policy map that gives all optimal actions in each state of the environment (Jordan, 2017). An example of a Markov decision process is shown in below figure (Fig. 4.19).

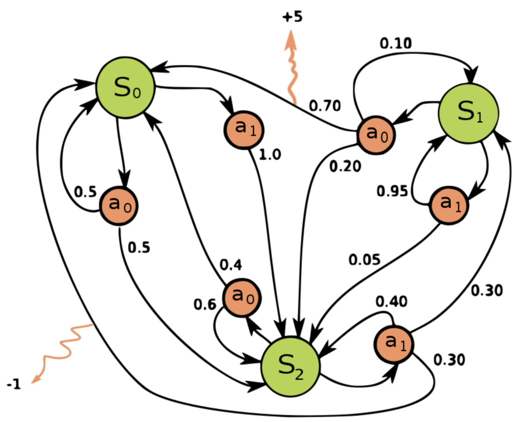

Fig. 4.19: A Markov decision process with three states (S), two actions (a) and two rewards (orange arrows) (Alvarez, 2017)

A policy that specifies the available actions at each state of the Markov decision process is binding upon the agent. The goal of the agent is to improve its policy to in order to maximize its gain or expected future rewards. Many algorithms of reinforcement learning learn from sequential experience, e.g. capturing the gain of a certain action in a certain state following a certain policy (Heinrich & Silver, 2016).

An agent is starting in the initial state S_0 and has two actions to make, with the transition probabilities of each action shown. The probabilities of all outgoing arcs of the actions sum up to 1. From state to state, by whatever action made, either a reward or a punishment is granted (Lozano-Pérez & Kaelbling, 2002).

In 2013, the London-based startup DeepMind published a paper where they presented a reinforcement learning algorithm that learned to play several Atari 2600 games by observing the screen pixels and receiving reward when the game score increased. In three of the games, the algorithm even surpassed human experts. It has to be noted, that the model learned from nothing but the video input, the reward and terminal signals, and the set of possible actions (Mnih, et al., 2013; Matiisen, 2015). To get a better sense of the kinds of games played, below figure is given (Fig. 4.20).

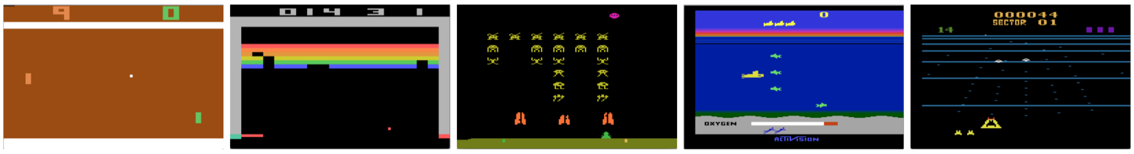

Fig. 4.20: Five Atari games' screenshots in gameplay mode: Pong, Breakout, Space Invaders, Seaquest, Beam Rider from left to right (Mnih, et al., 2013)

For this, the team has combined deep learning and reinforcement learning to something called deep reinforcement learning, to gain the positives of the large data handling of deep neural networks. The raw image data goes into a CNN with three hidden layers that result in a fully connected output layer with a single output for each valid action which were between four and 18, depending on the game. The outputs correspond to the predicted Q-values of the individual action for the input state. In Q-Learning, a method of reinforcement learning, Q-values are values in the Q-matrix, representing the agent's memory of what was learned by experience with the rows of the matrix representing the current state of the agent, and the colums the possible actions leading to another state (McCullock, kein Datum). Mnih, et al., refer to this approach as Deep Q-Networks (2013).

DeepMind's breakthrough in developing a general gaming AI that needs no further adjustment or specialization for other games in their research other than game-specific preprocessing and cropping of the visual input (ibid.). The industry's attention led to further research in this field. A paper published by the University of British Columbia deals with learning physics-based locomotion skills, e.g. walking of a three-dimensional multi-joint body on a map, via reinforcement learning. The aim, and reason why reinforcement learning was used to train the model, was to learn a variety of environment-aware locomotion skills with limited prior knowledge (Peng, et al., 2017). Similarly, autonomous vehicles also benefit from deep reinforcement learning frameworks (El Sallab, et al., 2017).

4.2 Future of Deep Learning Research

What research in the field of deep learning as a whole brings, is unpredictable. Deep learning has gained much traction over the last few years, although the concepts were not completely novel (see Chapter 2). In the case of a standard (or vanilla) neural networks, the model of perceptrons were known long ago, strongly oriented to neurons in the brain. Also, the visual cortex laid the ground for convolutional neural networks, whereas some connections of GANs are found in game theory developed fifty years ago (Wang & Raj, 2017).

Although deep learning has shown tremendous progress in applications of artificial intelligence such as speech recognition, image detection and prediction of future events, there are shortcomings when it comes to learning like a human does. In the 2015 paper of Lake, et al., the modern machine and deep learning models were challenged to learn new concepts from just a few examples, calling this *few-shot learning*. For a human, it would be easy to identify a new two-wheel vehicle from one picture, whereas a machine would need much more examples. Meta-learning, or *learning to learn*, has recently become a hot topic, mainly to raise neural networks and artificial intelligence to a new standard (Finn, 2017). Meta-learning defines the ability to learn new tasks after being exposed to a large number of tasks, e.g. classifying a new image within five classes on basis of training on one example of each class (Ravi & Larochelle, 2017). The idea of meta-learning and learning from a few examples was recently researched further by a research group of Huawei (Zhou, et al., 2018). Zhou, et al., proposed a *deep* framework for meta-learning: a combination of concept generator, coupled with a concept discriminator and a meta learner. Other concepts of meta-learning include fast reinforcement learning, and hyperparameter optimization (Finn, 2017). Hyperparameter values are set before the training of the model, including number of filters, dropout rate, learning rate, cropped image size, and layer type and size (Miikkulainen, et al., 2017).

With the rise in computing power possibilities and the shrinking costs thereof, older concepts should be revised again from a different viewpoint. The constant review of older concepts might uncover unknown effects. The following paragraphs therefore deal with older models, with yet unreached goals.

Amongst the ideas of a more realistic, or nature-true, modelling of neural networks, the concepts of spiking neural networks (SNN) and hierarchical temporary memory (HTM) appear to be promising. Spiking refers to the short and sudden increase in voltage to send information throughout the biological neural network. Recent research has shown, that neurons not just encode information in their average firing frequency but also in the timing of the spikes. This pulse coding is thought to be computationally powerful and promising in tasks where temporary information need to be processed (Vreeken, 2003). Hierarchical temporary memory is defined as a *machine learning technology that aims to capture the structural and algorithmic properties of the neocortex*. Being the base of intelligent thought in mammalian brain, the neocortex combines vision, hearing, touch, movement, language and planning in one place. However, there are no specified task centers for each of these *senses*. Biological evidence suggests that the neocortex implements a common set of algorithms, designed to perform many intelligent functions. Following the model of the neocortex, an HTM network is stacked in several layers, communicating within and between levels as well was outside the hierarchy. This is hoped to be of importance for multisource sensor data and the like (Numenta, 2011).

Another approach is the concept of imagination machines. Current achievements of deep learning are based on learning patterns and probability distribution from data, whereas imagination machines, imagination defined as *the capacity to mentally transcend time, place, and/or circumstance*, is more than that. Imagination science, according to Mahadevan, addresses the problem of generating samples that are novel, not from the training. This might be art, linguistics, e.g. new or unknown metaphors, or (intuitive) new ways of problem solving. Although this field is relatively new, the need for an agile AI is definitely existent (Mahadevan, 2018). The previous concepts presented in this subchapter might be of help to achieve some traction in this still very theoretical field of imagination machines.

There are also thoughts of quantum-based neural networks, or quantum neural networks (QNN). The quantum neurons, also called qurons, would then act as a two-state (resting and active) logic operator. Yet, the work done on this subject is very theoretical and does not go beyond idea suggestions and theoretical considerations rather than a fully functional model (Schuld, et al., 2014).

No doubt, the next big leap to reach is Artificial General Intelligence (AGI), e.g. an AI that can perform a universal intellectual task that can originally only be done by a human. The concepts in this thesis would most likely be defined as *narrow AI*, e.g. systems that carry out specific intelligent behaviors in specific contexts, according to Ray Kurzweil. If one changes the context or the behavior specification slightly, some level of human reprogramming or reconfiguration is generally necessary to retain the level of intelligence of the machine (Kurzweil, 2005). Natural generally intelligent entities like humans, having a broad capability to self-adapt to changes in their goals or environment, perform transfer learning to generalize knowledge from one context to others. Artificial general intelligence therefore has emerged as an antonym to the term narrow AI (Taylor, et al., 2008). Fundamental requirements, according to a 2017 paper by Facebook AI Research, are that a machine should be able to communicate through natural language, having a *learning to learn* ability, e.g. transfer learning, being able to learn without an explicit reward score, e.g. solely through linguistic feedback, and a general interface with no need for manually re-programming when applied to another domain (Baroni, et al., 2017).

Another concept that is fueling artificial general intelligence is transfer learning. A paper from DeepMind outlined that it would be efficient for AGI if several users *trained the same giant neural network, permitting parameter reuse*, and proposed a framework called

PathNet. Such a framework can help networks reuse existing knowledge instead of learning from scratch for each task. DeepMind has shown this concept to work with several Atari games While some games could not profit from transfer learning, e.g. the hyperparameter search for the network took longer with PathNet than manually, some games' transfer score (the area under the learning time, see Fig. A.1) could be multiplied (Fernando, et al., 2017).

Although some concepts discussed remain vague and very theoretical without any sound implementation or repercussion in academia or industry like HTM and imagination machines, further research might prove one or another to be a helpful tool in achieving better results, accuracy and broadening possible tasks in AI. Especially better technical hardware and architecture, with special focus on quantum computing and AGI, can make unfeasible concepts of today work well tomorrow.

4.3 Obstacles

Naturally, deep learning has limits thus far, like every major complex technology. A paper from the New York University points out theses about the limits of deep learning (Marcus, 2018). Some are outlined as follows:

Deep learning thus far

- is data hungry
- is shallow and has limited capacity for transfer
- has no natural way to deal with hierarchical structures
- is not sufficiently transparent
- cannot inherently distinguish causation from correlation
- presumes a largely stable world, in way that may be problematic
- works well as an approximation, but its answers often cannot be fully trusted
- is difficult to engineer with

And of course, these limits are part of ongoing research and constantly gets pushed further with new inventions and novel findings that lead to even more research, as it is the case with transfer learning, novel architectures with less input data needed, and research to biased input data (Tommasi, et al., 2015; Sawada, et al., 2017; and see subchapters 4.1.4, 4.1.5, and 4.2).

In many industries, a simple black box model is not sufficient. To get behind the decision-making of a machine, the underlying algorithms and mathematical models are ought to be known, but even then, for solving complex problems with the help of deep neural networks it is mere impossible to deliver a causal explanation (Stövesand, 2017). That this cannot be the case, and a machine's reason has to be transparent to humans, is widely expected in politics, too. In mid-2018, the European Union may require, that companies should be able to give an explanation for decisions that automated systems reach. Considering that even the developers and developing engineers cannot understand the reasoning of a machine's decision that they have built themselves, this might be a huge obstacle.

As described in subchapter 2.5, in certain circumstances, a machine might or should not be able to make decisions when it may cause in fatalities. Recently, autonomous vehicles have made negative headlines in some fatal cases. The most recent one, an Uber autonomous vehicle crashing with a bicycle rider on the night of March 19^{th} of this year, resulting in the cyclist's death, although a human supervisor was behind the wheel, has caused controversies. While the in-car camera footage shows the human supervisor not looking on the road or being distracted moments before the crash, the car itself did not manage to break promptly. Recent investigations have shown that the crash was a result of a *false positive*, e.g. deciding it might be a cardboard or something else, that would not be necessary to break the car for, while in reality it was a case to break (Coberly, 2018). This kind of negative publicity makes it harder for artificial intelligence and deep learning to be included in everyday life of humans, especially if it means the possibility of fatalities.

4.4 Market Intelligence

In many industries the artificial intelligence and deep learning is being reviewed thoroughly for new business applications and the use of it to upgrade current products and services. Tractica, a provider of market intelligence and research, published a market projection for the years 2016 to 2025 with a cumulative market revenue of approximately $63.1 billion for different fields of artificial intelligence (Fig. 4.22). Since deep learning has already shown to outdo former classical machine learning algorithms in the fields of speech recognition, object recognition and detection, pattern recognition, and prediction of the future based on the past, it is believed that the bulk percentage of revenue would be on the account of deep learning applications. In fact, a survey published by the Bank of America Merrill Lynch in 2015 showed that half of AI revenue is induced by deep learning ($1.05 billion). Another survey on the same subject showed that by 2025 the share of deep learning of the total AI revenue will grow to about 73 percent ($93 billion). The cumulative total of artificial intelligence revenue adds up to $127 billion according to that study (Ma, et al., 2016, p. 14), although all in all the numbers and percentages should be treated with caution.

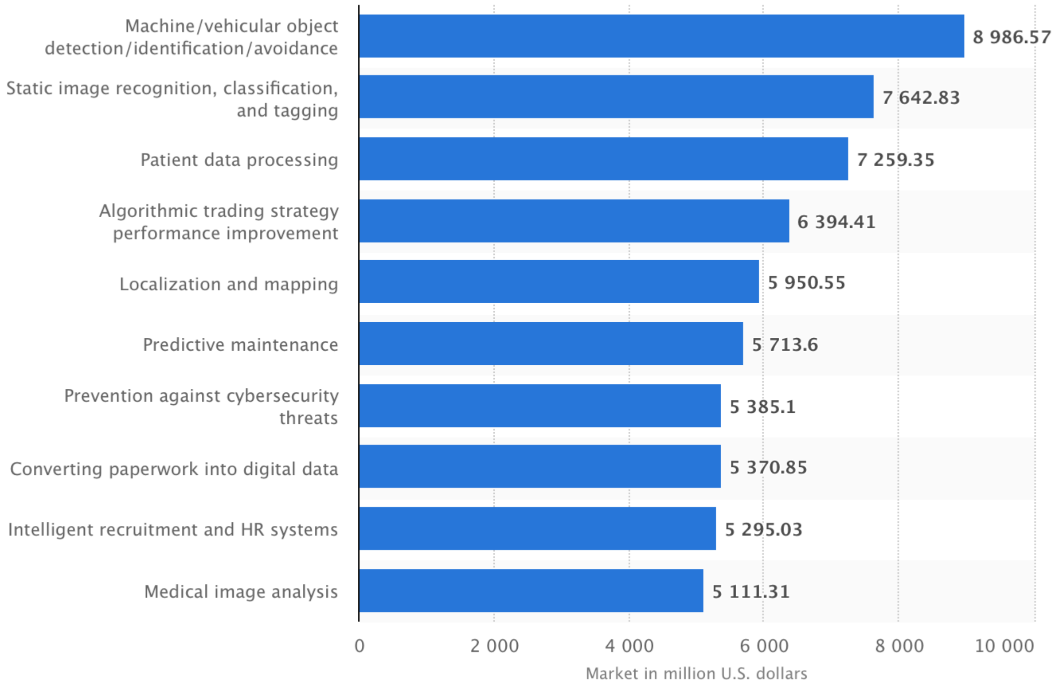

Fig. 4.21: Cumulative revenue of top 10 segments of AI markets worldwide between 2016 and 2025 (Statista, 2017)

5 Deep Learning Use Cases

The problems addressed today by deep learning in Chapter 4.1 were described from a more general perspective. This chapter deals with specific use cases of deep learning in the medical industry as well as in the aviation industry. As the first theoretical chapters made it clear, deep learning is a general computer science technology, which is being used in many industries. The specific circumstances in every case makes it difficult to generally apply applications to another problem in another field. In the example of computer vision and object recognition, preceding steps need to be considered properly. Datasets for the training of the neural networks have to be prepared carefully and thought out, sorted, cropped, and preprocessed to ensure optimal results. Although products and services can be enhanced with deep learning and the use of artificial intelligence, the problems that occur in real-world applications are to be taken into account, whether they are of an ethical, political, environmental or economical nature.

Another issue that has to be pointed out is the fact, that use cases in artificial intelligence and deep learning are interdisciplinary in nature. There are strong bonds to big structure-related topics, such as big data, ontologies and knowledge bases that build up to interdisciplinary fields like knowledge mining, pattern recognition and automated diagnosis (Ma, et al., 2016, p. 14).

At the time of writing, many use cases are being discussed in both the medical and aviation industry. The following subchapters shall give a better understanding of the two industries dealing with deep learning to enhance their current business and future business opportunities. It has to be understood, that the small number of five reviewed use cases per industry do not represent the needs of the industries as a whole, regarding deep learning and artificial intelligence. In contrast to traditional machine learning, deep learning is designed to learn more from extensive and complex data, and to adapt to unforeseen circumstances. This makes it to be on the cutting edge of artificial intelligence technology as of today.

5.1 Medicine

The medical industry is classified by the International Standard Industrial Classification into three subcategories:

- Hospital activities
- Medical and dental practice activities
- Other human health activities (not performed by hospitals or medical doctors)

This also includes pharmaceutical, bioengineering and medical engineering activities. The main goal is to maintain and improve human health conditions and to treat and battle diseases of all kind. Deep learning is therefore a tool to enhance the medical service in hospitals and clinics, as well as to elevate the understanding of diseases and improving related therapies. In some cases, the problems are addressed by hand, e.g. detecting lung cancer from computer tomography scans, and in others, the sheer amount of data is not or with a lot of difficulties evaluable, e.g. recognizing rare diseases from large databases of medical data. The following use cases are being researched and, in some cases, already partly implemented in the medical industry.

5.1.1 Use Case I: Predictive Diagnosis

For a human to learn diseases and to predict when they show up in a patient, is a tough task. However, medical data has been recorded since the digitalization of health records and applying learning algorithms seem to be the only option to make predictions based on data. This was also the reason why Sutter Health and IBM tried to apply artificial intelligence and machine learning on the data to gain more insights about the influence of certain circumstances in a patient's health record on heart failures. For that, about 30,000 health records were combed through and analyzed with big data methods, machine learning and natural language processing. Natural language processing was used to analyze and extract information out of the physician's notes and match it with the rest of the patient's record. The sorted data was then analyzed with different methods of machine learning, namely logistic regression, support vector machines, and n-nearest neighbor (Barrett, 2017). Although the results were promising, the feature extraction had to be done by hand. Choi, et al therefore applied deep learning to this problem and could increase the prediction rate up to 81 percent. While logistic regression and the support vector machine were predicting slightly lower when giving an input of 265,000 patient's health records, the multi-layer perceptron model was improving in performance up to 82 percent (Choi, et al., 2017). While the old guideline for predicting heart failure in patients included 28 cardiovascular risk factors, only six were consistently found to be predictors of a future diagnosis of a heart failure. According to Stewart, one of the authors of the paper, it is possible to predict heart failures one to two years earlier with the new model (Barrett, 2017).

In another case, Google has made headlines about predicting cardiovascular risk factors from retina photographs via deep learning. About 285,000 images were used to train and develop the deep learning model and about 13,000 images to validate it. Retina fundus heat maps were analyzed with a convolutional neural network, after the features and feature areas were defined by three ophthalmologists on a number of 100 retina heat maps (Poplin, et al., 2018, p. 158 ff.). The new method has shown to predict a heart failure with an efficiency of 70 percent, in comparison with 72 percent of the old SCORE method which requires a blood sample, though (Vincent, 2018). Below figure shows the heat maps of each prediction (Fig. 5.2).

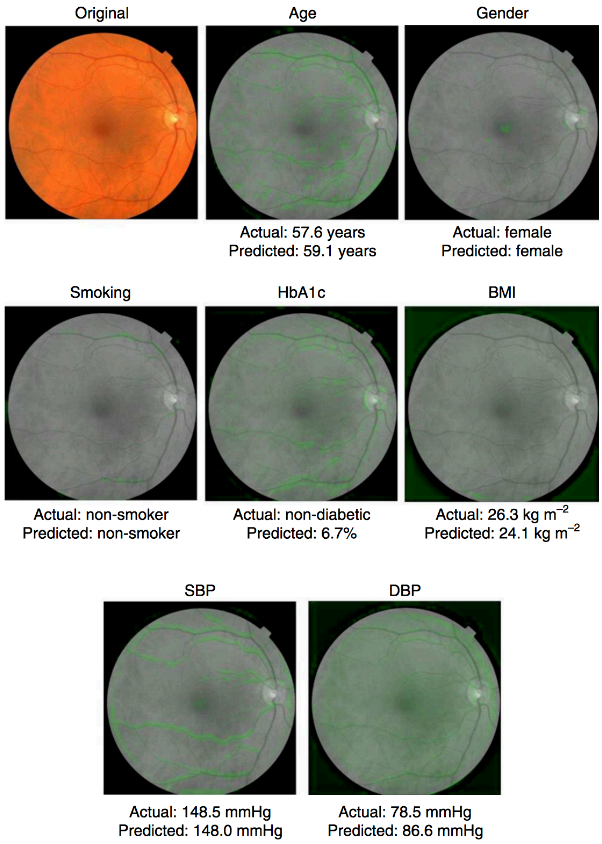

Fig. 5.2: Sample retina image in color (first row, first from left), remaining images gray scale with each prediction overlaid in green (Poplin, et al., 2018, p. 162)

5.1.2 Use Case II: Medical Imaging

Skin cancer is one of the most cancer cases in the western world, largely due to ultraviolet light emission, with about 5.4 million new non-melanom and nearly 200,000 melanom skin cancer cases every year in the United States alone (Skin Cancer Foundation, 2018). Detecting it happens visually by medical experts with the naked eye and with the aid of a dermatoscope, which is a handheld microscope. If the first visual examination makes the dermatologist to believe it is cancerous, the next step is a biopsy. Detecting a cancerous melanoma in one of the earliest stages, the five-year survival rate is 97 percent, in contrast to 14 percent if it is detected in its latest stages.

A research group of the Stanford Artificial Intelligence Laboratory came up with the idea to detect presumed melanoma spots on a patient's skin with machine vision. Training the deep convolutional neural network with nearly 130,000 samples of about 2,000 skin diseases, it demonstrated promising accuracy from the very first test (Esteva, et al., 2017; Kubota, 2017). The sample data was extracted from the internet through search engines and the test data came from the University of Edinburgh and the International Skin Imaging Collaboration Project. Putting the algorithm into test, it matched the performance of 23 board-certified dermatologists. Thinking a step further, implementing it in smartphones, the lives of millions or even billions of people could be saved by letting a deep learning algorithm classify the taken images (Kubota, 2017).

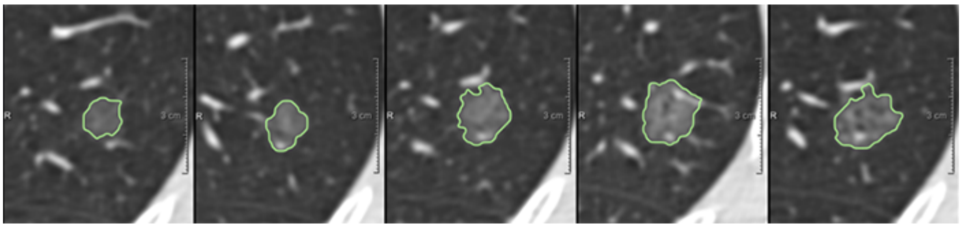

Fig. 5.3: Series of five CT scans with an automatic detection of a lung tumor (DIAG, 2013)

In several other papers, detecting lung cancer on a CT scan were described as still challenging. Although the use of 3-dimensional convolutional neural networks shows some advantages, the main focus lies on improving the training datasets. The goal is to shift from the current procedure, where a series of CT scans lead to a somewhat certain statement of the radiologist, to a one-time CT scan with a deep learning system classifying possible tumors (Kuan, et al., 2017; Chon, et al., 2017). Above shown figure shows CT scans within a period of three years and an automatic object detection with the help of learning algorithms (Fig. 5.3).

5.1.3 Use Case III: Personalized Drugs and Therapy

The usual appointment at the doctor's practice is more about the condition the patient has than *who* the patient is specifically. Therefore, medical treatments most often aim to cure the general person and if a treatment does not work, another, still promising method will be used in the process to heal the patient. Despite high chances of a positive treatment, there still is a possibility that prescribed drugs do the exact opposite: act toxic. Depending on many factors, fatal drug toxicity can be linked to the patient's age, genetic makeup, preexisting conditions, the dose of the prescribed drug, and other drugs that the patient may be taking (Sarmah, 2014). A study showed that more than 3 percent of deaths in Sweden are due to bad reactions to pharmaceuticals. In the US these fatal adverse drug reactions as they are called, account for 5 percent of death numbers (Cressey, 2008).

What is happening since quite some years, with sinking prices of genome analysis and evolving data landscapes, is the forth bringing of several new approaches of personalized medicine to combat the unwanted effects of drug reactions. However, not only the prevention of unwanted effects is part of the research, also wanted response of specific cells to a new drug or new treatment is being in the focus. As the National Human Genome Research Institute defines it, personalized medicine includes *an individual's genetic profile to guide decisions made in regard to the prevention, diagnosis, and treatment of disease* (McMullan, 2014). Although the obstacles to introduce a new drug are high, ways have to be found for testing. Occasionally, when the physician in charge has a hint of a certain effectiveness of a drug or the weak point of a disease, e.g. cancer cells, in most of the cases, the physician is able to initiate special treatment and genetic testing. As of 2014, researchers have discovered several thousand disease genes and have access to more than 2,000 genetic tests for human conditions. Also, as of said date, more than 350 biotechnology-based products were in clinical trials. One of the most advanced areas in terms of personalized medicine is lung cancer treatment with several FDA-approved drugs and biomarkers in clinical trial (McMullan, 2014).

So, based on the genomic fingerprint, new types of drugs and therapies are on the rise that directly and precisely address the specific aim. Deep learning in this matter is a solution to the problem, for genomes produce large amounts of data which is an impossible task for a human to evaluate and find patterns. Generative models (see subchapter 4.3) make it easy to simulate new drugs and test the effects on human pathology before going into clinical trial. This field remains highly interesting and promising as well as competitive in terms of research and publications, since generative models are lately making encouraging progress and showing value in real-world applications (Machart, 2018).

5.1.4 Use Case IV: Image-to-Speech Aid for the Blind

Around 253 million people worldwide live with vision impairment. Of those are 36 million blind and 217 million with moderate to severe impairment, as a study of the World Health Organization shows (WHO, 2017). Tasks, such as crossing the road, buying groceries or recognizing faces, become an everyday struggle.

A Milan-based startup, *Eyra*, is trying to change that. Their product called *Horus*, a headphone paired with two cameras, uses computer vision and deep learning to process the environmental video input and gives an audio output (Fig. 5.4).

Fig. 5.4: Horus, a dual-camera upgraded headphone with NVIDIA Tegra K1 GPU (Disup, 2016)

In detail, Horus can detect everyday objects and people and assists the user via extensive audio cues. A three-dimensional sound environment with different intensity, pitch and frequency feedback helps navigating the user and representing obstacles. Additionally, a short audio description of what the cameras are seeing will be given out.

With the help of cheaper and more compact GPUs and camera systems, the combination of natural language processing, image detection and recognition, this might be of help for those who are dependent on aid. Especially those who did not had the chance of learning braille or the keyboard layout, or not getting familiar with the use of classical user interfaces, this is a good chance to keep up with new technology waves and the social media evolution (Excell, 2016). Understandably, this kind of product is utmost complex since it deals not only with a simple task for an AI to solve, but an interlocking stack of complex tasks that need consideration of further concepts. This would be beyond the scope of this chapter since the main example (Chapter 7) is another use case.

Either way, machine vision can also be used in other cases for augmenting the human vision of the user, e.g. real-time three-dimensional projection of a tumor in a patient's body with an augmented reality headset for medical training purposes.

5.1.5 Use Case V: Behavioral Modification

With the occurrence of micro sensors and easy-to-use applications for personal data handling, new ways of using that kind of data arises. Users are much more willing to gather data, if there is a clear personal advantage coming with it. The numbers speak for themselves: An estimated total number of 33 million sold Apple Watches by end of the third quarter of 2017 and a roughly annual growth of 50 percent (Heisler, 2017).

On the other side, according to the Center for Disease Control and Prevention around 7 million people die from smoking-related diseases every year in the world. Around 890,000 deaths are the result of exposure to second-hand smoke (WHO, 2018). Lung cancer is said to be one of the most common types of cancer. To battle the high numbers of cancer and smoking-related deaths, a young startup called Somatix is on a mission to help change smoking habits. They developed an application that works with various fitness trackers and smartwatches and uses the built-in sensors to detect hand-to-mouth gestures, which indicates smoking. This cognitive behavior therapy, as they themselves call it, analyzes the sensor data with machine learning to passively monitor the state of being and deliver insights to the user (Toren, 2016). This can most effectively be achieved with LSTM networks (see subchapters 4.3 and 5.2.1).

Although it is expected to further dim the use of tobacco for those, who are willing to change their behavior, it must be said, that 80% of the world-wide smokers live in low- and middle-income countries. Most probably those are not the usual customer base for fitness trackers or smartwatches, which make it hard to battle smoking related diseases overall (WHO, 2018).

It is possible to adapt this concept to other behavioral problems, for example bad posture correction. With the help of sensors and cameras, an AI detects bad postures and either sends a message to the person's phone or, in a more connected future, sends haptic signals to the person's accordingly equipped clothing.

5.2 Aviation

The aviation industry deals with aircraft and aircraft related products, components and services on a passenger's flight journey and throughout the aircraft's lifecycle. This includes the manufacturing, development and design, repair, maintenance, overhaul, operation and use of aircraft and aircraft components, as well as the operation and coordination of flight management facilities and airports. Usually, it is being distinguished between civil and military aviation.

This chapter deals with use cases from civil aviation and in particular with the repair, maintenance and overhaul fields of aircrafts. However, for a better understanding of the actual technology behind the use cases, examples from other industries might be given at some points.

The described use cases were all being handpicked and filtered through a number of interviews with several aviation experts, both engineers and managers.

5.2.1 Use Case I: Predictive Maintenance

Aircraft regularly visit maintenance hangars to get inspected. The checks are being performed with a differing period in between. Also, the time and tasks for the checkup differ according to the maintenance instructions. An aircraft is checked regularly every 400 to 600 flight hours, or 200 to 300 cycles, which makes an A check binding approximately every month. Every six to eight month a B check is binding, although many maintenance facilities tend to split the labor into A and C checks. C and D checks are performed at greater intervals, every 20 to 24 months, respectively every six to ten years. D checks are also known as heavy maintenance visits, which can take up to two months and at which the aircraft gets completely dismantled and inspected thoroughly, including a complete paint job. This happens about three times in a lifetime of a regular commercial aircraft and cost several million dollars, depending on the aircraft's model (Ackert, 2010). However, there exist other maintenance programs which split certain maintenance tasks (C-Light / C-Heavy) into shorter or longer periods, respectively (Ackert, 2010).

A 2015 study showed that 8,500 aircraft have been retired until that date at an average age of 27.2 years. Although it is possible to stretch this time to a longer period, like it is usual in the military with enhanced maintenance programs, up to double the normal period (Haggerty, 2004), the retirement decision is based upon maintenance and operating cost versus the financial contribution (Forsberg, 2015). Below figure shows the maintenance checks throughout an aircraft's life (Fig. 5.5).

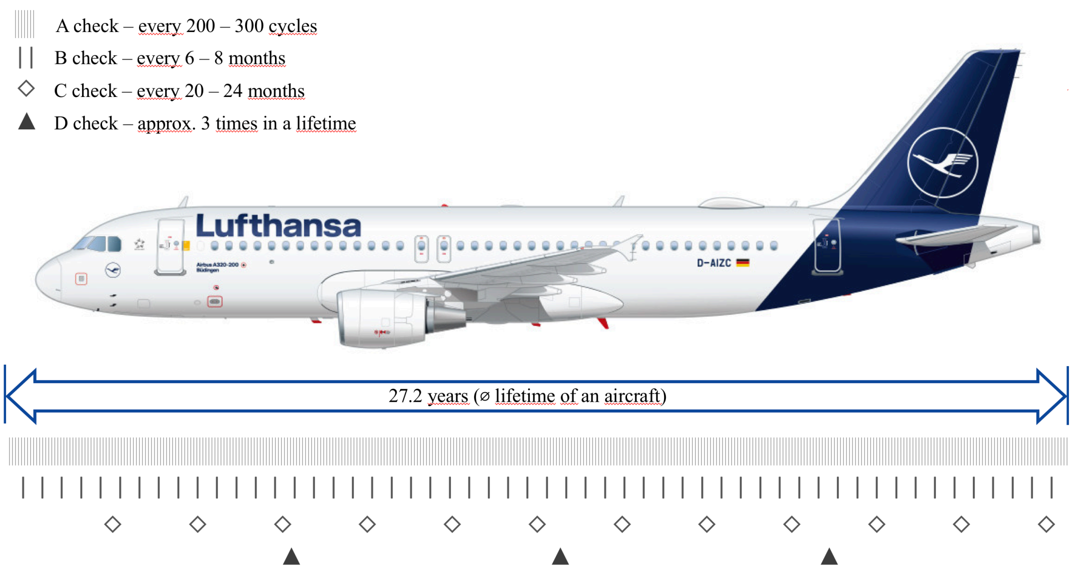

Fig. 5.5: Maintenance program with all checks throughout the lifetime of an aircraft; own figure, image of the Airbus A320-200 from (LH, 2018)

To better plan maintenance inspections and part replacements, and thus minimize operating costs, the concept of predictive maintenance is being discussed heavily in the last years. There exist about four major maintenance types: reactive, preventive, predictive and proactive maintenance. The main difference between all these maintenance types is that *reactive* follows a run-to-failure strategy, which is only possible for non-safety related components of an aircraft, whereas *preventive* aims to maintain machines and components at specified time intervals, even if that would mean to replace good components (see Fig. 5.5). However, in the long run, it pays off since the risk of machine failure occurrence is being reduced significantly (UE Systems, 2014). To design economically viable maintenance programs, it requires experience in the field and a good risk management to calculate the optimal and economically reasonable checkup intervals.

Predictive maintenance on the other hand means routinely inspecting machines and components while operating, with the help of external sensing technologies such as heat and pressure sensors to track lubrication wear and tear, for example. With this, detecting faults and parts breakdown in machines becomes possible, which would otherwise be very difficult (UE Systems, 2014). Spontaneous machine and component failure would occur more seldom and could help planning maintenance programs more efficiently. Besides process efficiency and heat loss, five nondestructive techniques are being used in general for non-mechanical equipment and components: vibration monitoring, process parameter monitoring, thermography, tribology and visual inspection (Mobley, 2002, p. 5 f.).

Proactive maintenance however tries to find out the root causes of failure. Although this means to keep track of the quality of products used, e.g. lubricants, or training the maintenance technicians, it can also go back to the machine's design (Casey, 2013). For an aircraft operator, this could mean flying on south-east Asian routes might result in stronger corrosion of unprotected metal parts of components due to salt, humidity and heat as root causes (EAI, kein Datum).

When talking about monitoring machine and component condition for predictive maintenance, deep learning is a key technology to make predictions based on time-series events and anomaly detection. While predictive maintenance models using traditional machine learning algorithms are based on feature engineering (see subchapter 3.3), which itself is constrained by the domain expertise and the high-grade of manual labor, and are hard to reuse since the extracted features are problem-specific, deep learning has shown great results automating the data and feature extraction. Among the used networks are *Long Short-Term Memory* (LSTM) which are a special kind of RNN and suited for learning from sequences over time. The data is collected over time to monitor the state of the component or machine with the goal of finding patterns to predict failures. With this technique, failures of engines for example can be detected before they occur, so that the maintenance can be planned in advance (Uz, 2017).

Baker Hughes has positively implemented predictive maintenance measures using deep learning on their gas and oil extraction equipment to monitor the pumps' health (Fig. 5.6).

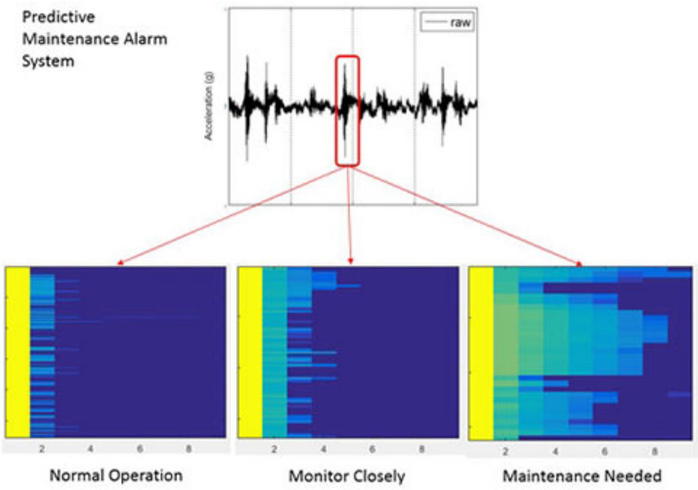

Fig. 5.6: Baker Hughes' MATLAB-based predictive maintenance alarm system (DE, 2017)

To train the neural network, field data of almost one terabyte was gathered, including temperature, pressure and vibration into MathWorks' MATLAB. Although preprocessing the data was a difficult task, since there were large movements of the truck and fluids, it could be automated using scripts to detect smaller changes of signals. Baker Hughes' savings add up $10 million through the use of deep learning in predictive maintenance (DE, 2017).

General Electric has experienced similar effects: Numbers of the GE Oil & Gas division show that the unplanned downtime rate was reduced from eight to 5.5 percent as well as the financial impact from $65 to $24 million dollars, when compared to preventive maintenance programs (Peeling, 2017).

For aviation, prediction based on deep learning could not just mean replacing parts before they break, it also can be an important way to predict non-normal behavior of the aircraft, including irregular piloting behavior. The flight recorder can be seen as a sensor, that gathers lots of data over time. Alternatively, other sensors that track position, altitude, rotation, sound and light, can also be applied to the cockpit to gather more exclusive data. Deep learning can help analyze this data and find patterns, compare them with findings from years before, and make predictions of possible states in the future. This would generate new knowledge in the field of complex machine behavior and human-machine interaction.

5.2.2 Use Case II: Visual Quality Recognition

In safety relevant environments like aviation, a high and steady state of quality is an important goal to achieve. This is not only relevant for maintenance and repair, but also in production. Sustainable product and process quality is a decisive criterion for a company's success and reputation. More than half of all product quality checks in production are based on the processing of visual information, e.g. an employee has to visually inspect at the product from every angle. This way, the finishing, seamless assembling and other quality criteria are met with the cognitive ability of the employee and his senses. And although the employees are highly trained, some optical damages are so small an eye cannot detect it. On top, this procedure is an exhausting, long-time continuous work and often depends on the mental state of the employee and his focus (Schröer, 2017). This also leads to reduction in production efficiency since faulty products result in sale deficit and a waste of raw materials (Wang, et al., 2017).

As it is the case with computer vision, deep neural network models like CNN's have shown to outperform humans in certain visual recognition tasks (He, et al., 2015). Visual detection of damages is therefore a promising use case in aviation, automotive, and in general manufacturing of safety relevant parts and products.

Wang, et al., have designed a, what they call, fast and robust CNN-based defect detection model in product quality control (2017). The convolutional neural network model was trained on a dataset from the German Association for Pattern Recognition and German Chapter of the European Neural Network Society which contains six image classes with the size of 512 x 512 pixel. Each class consists of 150 defectives with one defect region and 1000 defect free images. Below figure shows each of the six image classes (Fig. 5.7).

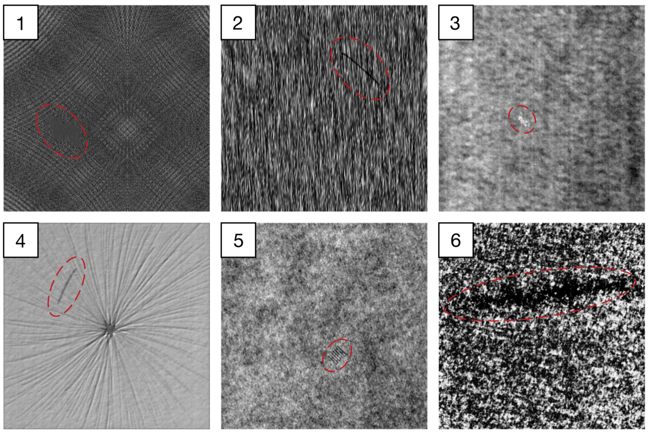

Fig. 5.7: Samples of each image class, which differ in background texture; the defective regions are marked by a surrounding red ellipse (Wang, et al., 2017)

Applying and training a 11-layer CNN network (see subchapter 4.1.2) was achieved with 70 percent of the dataset as the training set and 30 percent as the validation set. The convolutional neural network was built in a two-fold manner: the global frame classification part and the sub-frame detection part. The global frame classification part was designed to classify the image sample into the correct class, thus detecting the background texture features. The sub-frame detection part was designed to detect whether the certain defective region or damage was part of the sample based on the trained input data.

The concept of Wang, et al., achieved an overall 99.8 percent detection rate of faulty images, outperforming former best performance algorithms. With a detection rate of 27 images per second, real-time detection would also be possible.

In maintenance and repair of aircraft, this is a powerful method to detect defects which is an exhausting task for a human, and where the safety of the component or machine is crucial, e.g. engines. A single engine visiting the maintenance shop and getting dismantled completely costs around 3.5 million Euros. But this approach is not only for detecting damages or quality shortcomings in machines and machine components, but also for other industries, that are dealing with classifying and sorting out products, e.g. tomatoes (Fig 5.8). In agriculture,

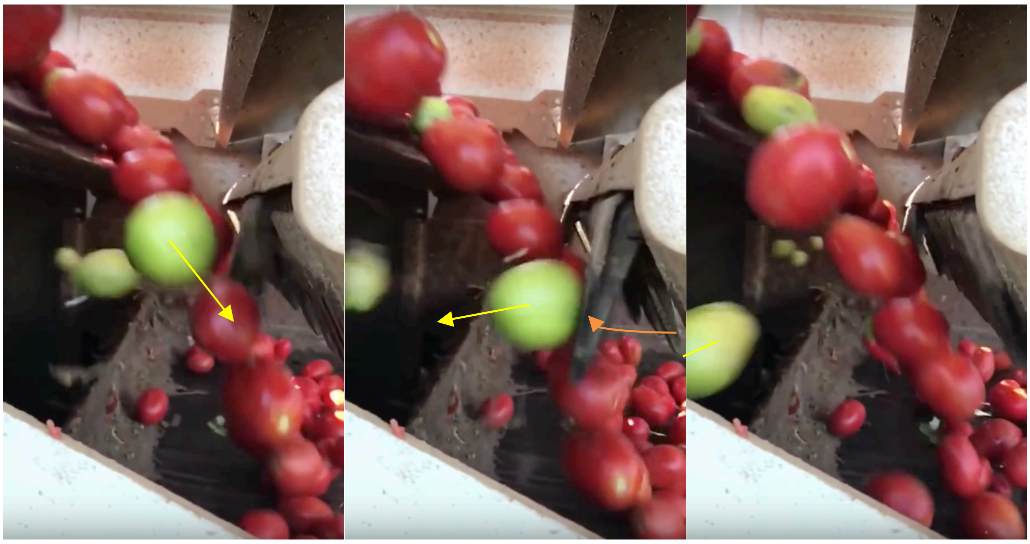

Fig. 5.8: Quality inspection: Tomato sorting machine based on computer vision and simple robotics; yellow and orange arrows show moving direction (Youtube, 2017)

In agriculture, there are several tasks that can be done by a deep learning computer vision machine. From detecting anomalies in crops, empty spots in the field, size and shape of the crop products, to parasites and other crop diseases, deep learning can be a powerful tool to detect and classify those specific items, and most importantly make decisions on basis of the digital image seen. Therefore, computer vision is an emerging subject in precision farming and food inspection (Zachevsky, 2012).

With regard to aviation and repair of components, decision making is also an important subject since there are other economical questions when repairing equipment. In most cases when maintaining an aircraft, some items need to be replaced, some can be repaired. Whether it should be repaired, or replaced, where it should be repaired (internally or external partners) or simply thrown away, are questions a maintenance expert has to deal with. In certain cases, though, the faulty item has to be analyzed further to make sure the optimal decision was made, regarding safety and finance. This is a step further in the making of intelligent machines, that can augment humans in this complex task. There would be much more data needed to develop an understanding of the lifecycle of an aircraft, the interplay of different components, and future economical as well as safe maintenance programs of aircraft.

5.2.3 Use Case III: Automated Claim and Contract Management

When it comes to contracts and negotiations, companies from every industry including aviation have to get professional legal help. Bigger companies have legal departments, where corporate lawyers assert the company's interests in negotiations. With value chains becoming more and more complicated and outsourced, this matter becomes more important. However, humans make mistakes and that is where an intelligent machine can be of supportive help. Especially in contracts, there might be phrases that are problematic because they can mean one thing but suggest another. This results in uncertainty and longer negotiations or even worse in not realizing certain aspects or changes made to proposals and thus leading to a bad financial situation regarding the contractual object. Although this matter is not just related to aviation, it can affect every interaction with manufacturers, industry partners and customers. With aircraft parts pricing in the millions, small changes or incorrect claims can have great impact on a firm.

A simple digitalization of the legal contract is not sufficient to transform the content into a structured commercial knowledge base. Data in documents and written text are usually extracted via text mining, a special form of data mining which itself is an analysis and extraction of structured data. For an intelligent machine to be able to extract patterns and findings, the data has to be prepared based on knowledge representation and knowledge reasoning. A contract document has metadata, e.g. supplier name and identification, legal entity identification, contract date, etc., but not the knowledge that is hidden inside it. To derive this key intelligence, clause libraries have to be built that include metadata which describe business-relevant characteristics of the clauses (Mitchell, 2017). The next step would be to prepare the data for the machine to learn from it. That means to connect meaning to the clauses in the library. To train a machine, lots of data would be needed, i.e. a lot of contracts need to be digitalized and brought into the specific form needed.

Since every company, customer and legal entity has their own style of writing, an artificial intelligent machine has to get through all the meaning in the text. This is done with Natural Language Processing (NLP), and the explicit application would be machine translation and question answering since juridical clauses have to be translated into easy to understand explanation. With this procedure, the legal department can be supported massively and critical clauses and changes to the document and its meaning can be outlined. NLP is also used in extracting the meaning of tweets using slang language and is doing so fairly well (Dyer, 2017).

With enough training data, including non-legal text mining, the machine can also gain the ability to build language itself. This is mostly done with RNN-based language models, where the probability of certain words and word structures are being compared with the learnt data (Jurafsky, 2016). Compared to common speech, this is but an easy task. Legal clauses are known to be both precise but also have different meaning, depending on the interpretation of the court and lawyers. Despite the difficulty of implementing a self-scribing legal contract machine, in matters of claim management, this could be of a great support to lay out certain aspects, e.g. false commitments made or identifying warranty claims.

5.2.4 Use Case IV: Fair Market Price Prediction of Spare Parts

Another use case to implement deep learning is price prediction. In the maintenance, repair and overhaul of aircraft, spare parts are not only an expensive matter in some cases but also a scarce resource which need to be coordinated and ordered wisely. With lean production and the shutdown of large warehouses, spare parts also have to be managed accordingly. And in some cases, functioning parts are also being sold to other companies or aircraft operators. For this, a fair market price is being predicted on basis of yearlong experience in industry and market. It has to be said, that the prediction is not the most optimal price, since there are only a small number of parts dealers, depending on the specific aircraft's part, that can distort the fair price of it. Having deep expertise in the maintenance and repair industry is most often not enough leverage to have. On top and understandably, part prices vary at different times and places.

With deep learning, the prediction of prices and price trend is possible, provided there is an adequate data base. In stock trading, investment bankers analyze data like news, companies' history, industry trends, and more to make price predictions. A learning machine could easily output predictions based on past experience. These trading algorithms however, are the golden asset of investment companies, there is little known about how well they function. Similar to the use case described in subchapter 5.2.1, stock market predictions are also build upon LSTM networks to include the learnt of the past into predictions of the future. This includes all news and information, technical and fundamental indicators, technical chart analysis, etc. (Reid, 2014).

Furthermore, there is also an unsupervised approach using reinforcement learning to let the model train on its own. A positive outcome, i.e. making profit, would trigger a reward for the model to confirm certain strategies, while a negative outcome, i.e. losing money, would cause a punishment for the current state of the model (ibid.).

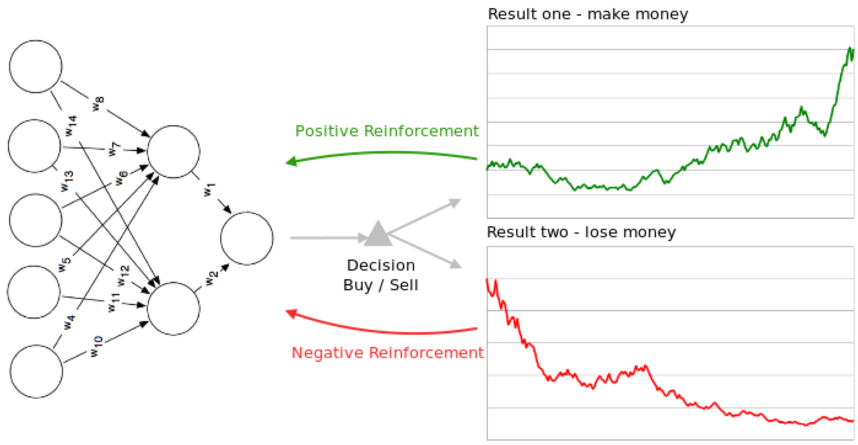

Fig. 5.9: Reinforcement learning: Making and losing money as reward and punishment, respectively, to adjust the weights of the neural network (Reid, 2014)

Just like DeepMind's Alpha Go model, a reinforcement learning-based market price prediction algorithm could start from zero. Since the algorithm of Alpha Go and Alpha Go Zero is too complex for a human to understand, chances are high that the outcome of such a self-learning trading algorithm would also be not understandable by humans with regard of how the prediction came into existence (see subchapter 4.3).

5.2.5 Use Case V: Natural Language Processing in Maintenance Records

Maintenance records are reports of aircraft going through the maintenance shop. This documentation process is more or less dictated by regulation authorities (FAA, EASA, CAA, etc.), binding upon maintenance and repair shops. In the Federal Aviation Regulations section 43.9, namely Content, Form, and Disposition of Maintenance Records, every *person who maintains, performs preventive maintenance, rebuilds, or alters an aircraft, airframe, aircraft engine, propeller, appliance, or component part shall make an entry in the maintenance record of that equipment.* This maintenance record includes a description of the work performed, the date of completion, names of other persons involved in the maintenance work, signature, and other information like certificate number and type of certificate (ASB, 2007). Usually, maintenance records are handwritten, and in most cases digitalized afterwards for an easier storage. But here exactly lies the problem: The handwritten reports do not share a common language or syntax. Each maintenance technician has their own style of writing, including the visual appearance of the written characters. This makes it hard to decipher for other maintenance employees for further documentation. Similar to the use case described before regarding automated contract management, to read and actually understand maintenance records, NLP is a solution to tackle this problem.

To address this problem, several steps are to be considered. First of all, the handwritten characters need to be recognized (see subchapter 3.2.3) which can be done with the help of deep neural networks (Fig. 5.10).

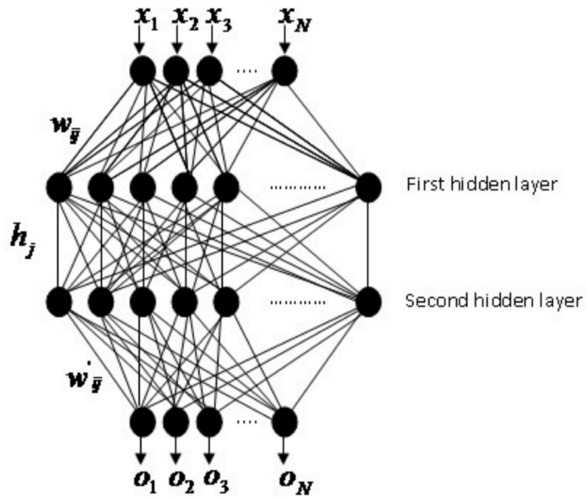

Fig. 5.10: Multilayer perceptron model with two hidden layers for handwritten character recognition (Kouamo & Tangha, 2012)

Although the proposed model by Kouamo and Tangha performed well with about 97 percent recognition rate, the training time was the longest of all researched methods (1h 32m). Other methods such as KNN can perform without the need of training time, but with a significantly lower recognition rate of 91 percent.

The next step would be probabilistic language models, as already discussed in subchapter 5.2.3. On basis of statistical models, every word has a specific probability, in comparison with other possible options. Given a good data resource, applying these models would not be that challenging, since there already exist proven methods to building probabilistic language models (Jurafsky, 2016). Put in simpler words, the recognized letters, words and finally sentences are being analyzed and matched with the most correct options, seen from a probabilistic perspective. On top, and in similarity with the legal clauses problem of subchapter 5.2.3, different words and word groups used by different technicians can be matched and generalized in a universal maintenance language. This would accelerate the maintenance process and prevent human error and bring maintenance shops a step further in the direction of true digitalization.

Extracting knowledge of handwritten maintenance reports is very similar to problems in other domains, like medicine. In medicine, healthcare records are also written by hand in most of the cases. A digitalization and automation of these reports can help nurses and practitioners and save precious time for more important tasks. Especially in this use case, a generalization of the technology used is intelligible.

6 Transfer of Technology

Development of new technology happens either at research faculties of universities or at R&D departments of bigger companies. The transfer of technology typically addresses the movement of academic discoveries and inventions into the commercial sector (Van Norman & Eisenkot, 2017). On the one hand transfer of technology for a large proportion builds upon intellectual property, that can be transferred from research faculties to commercial companies for royalties, on the other hand though, used in a cooperation and joint development and commercialization where both parties agree to certain roles in a development project. However, the subject of the thesis was aimed deliberately to act in an interdisciplinary setting between two different domains: the medical and the aviation industry. One guiding principle was the analogy between patients and aircraft, both suffering from diseases or failure in components respectively. Analyzing both industries in-depth showed that the matter is more complicated and cannot be represented by such a simple analogy. However, along the operational chain, from registering the current state, comparing with the target state, documenting, recommending approaches to reach a favorable outcome and to learn and adapt for possible future predictive measures, deep learning can be an enhancing technology in both the medical and aviation industry.

There also exist other types of technology transfer, between companies, between countries, and between companies and countries (Borge & Bröring, 2017, p. 312). This chapter however deals with transfer from academia to the commercial sector, i.e. private companies.

6.1 Overview of the Technology Transfer Process and Transfer Instruments

The process already begins with the disclosure of the invention or finding of a technological innovation. Although not all universities and research centers have a technology transfer office (TTO), around 84 to 87 percent cannot cover the costs of such an office, most often the supervisor of the research center undertake the task of technology transfer and commercialization together with the inventor (Van Norman & Eisenkot, 2017). However, due to the fact that the technology transfer office is being a home to the range of responsibilities in a technology transfer, further mentioning of technology transfer is linked to the TTO. Below is a representation of the technology transfer process given Fig. 6.1), however, this representation only covers the commercialization part of the transfer process, as described by Van Norman & Eisenkot. To avoid unnecessary duplications, further explanation is given in subchapter 7.4.2.

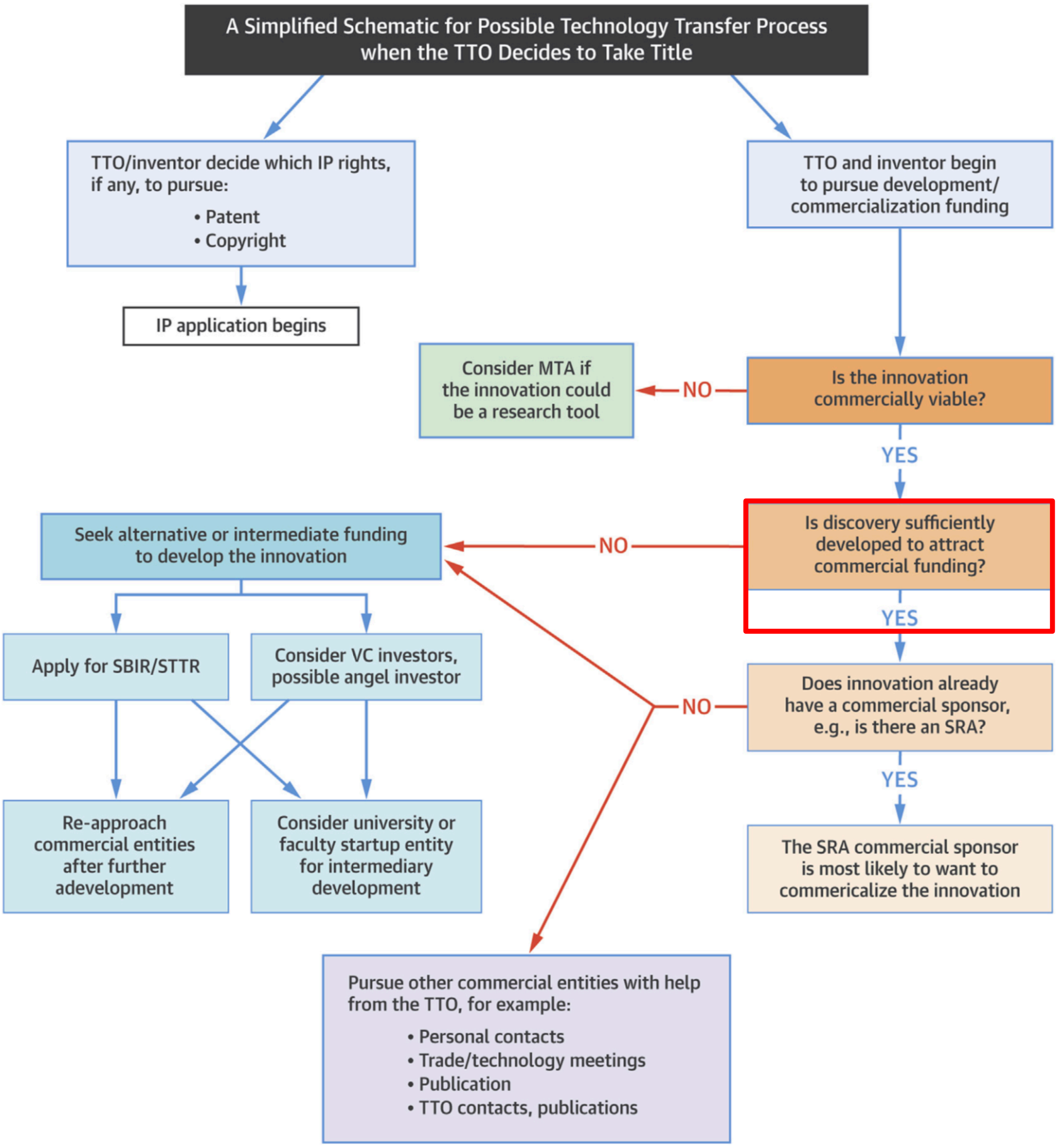

Fig. 6.1: Simplified overview of the technology transfer process (Van Norman & Eisenkot, 2017); Red box is additional and applies to findings in Chapter 7

Van Norman & Eisenkot focused their investigation on the process after the TTO decides to commercialize a certain technology, which at that time might still be in development. However, the finding of technology and the use thereof is also considered a step in the technology transfer process, as described in the following paragraph (see transfer instruments in Fig. 6.2). Having this in mind, the work of this thesis can also be called a part of the technology process. More specifically, this thesis work has great similarity with the Research Days mentioned in the below figure. Research Days can be described as a match-making process between a company and a research center, where the company (in this case Lufthansa Technik) makes a wish list from possible collaboration fields, and the research center (in this case the Institute of Medical Systems Biology) proposes relevant projects and use cases, and presents the chosen topic to the company after a preparation time of three to six months.

It is crucial for the transfer and also for the direction of development specialization to know what readiness level the investigated technology has (technology readiness is defined as the stage of development a technology goes through from basic research to fully viable commercialization; see subchapter 6.4.1 for a detailed explanation). The figure below shows the Helmholtz' application readiness level metric that shows large similarities with the technology readiness level of NASA (Fig. 6.2).

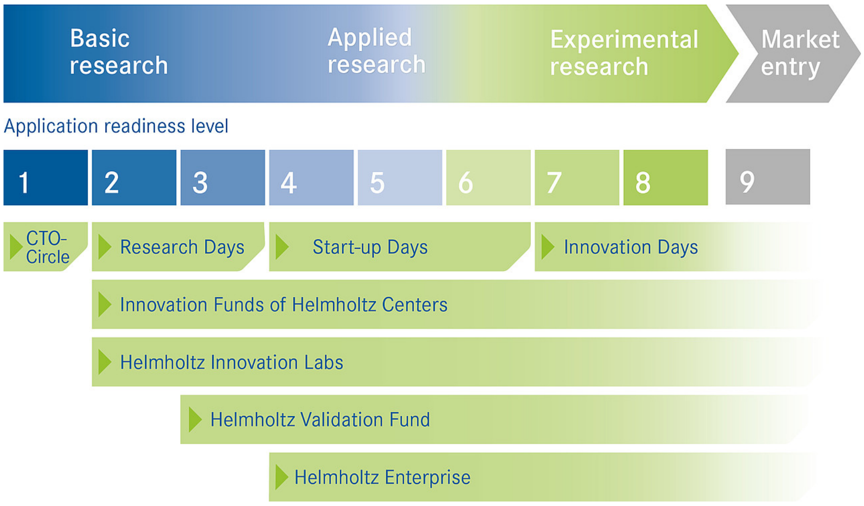

Fig. 6.2: Application readiness level of the Helmholtz Association of German Research Centers (Helmholtz Association, kein Datum)

In above figure, the possible instruments are shown which the Helmholtz Association use for transferring technology. The first row, CTO-Circle, Research Days, Start-up Days, and Innovation Days represent events or expert platforms where leading technology managers, technology experts and companies get invited to gain insights and initiate possible collaboration projects, as already explained for the concept of Research Days. The second row and the third row represent possibilities of research funding, being independent at the research center or as a more open format of cooperation, respectively. For advanced readiness levels (fourth and fifth row), assessments regarding a technology and market readiness or spin-offs take place, respectively (Helmholtz Association, kein Datum).

Further use of the last two figures are in subchapter 7.4.2, when applied to a specific transfer use case.

6.2 Requirements for Transferring Technologies

In many cases using and implementing a new technology is costly. The main reason for this is the lack of experience, that comes with new technology. Since new technologies and innovations are always a risky undertaking, many companies cannot include those kinds of innovations inside the predominant company structure (Mitra, 2013). Disrupting innovation therefore happens outside the main business divisions of a company. Dealing constantly with new shifts in the technology world, most companies have strategic scouting functions to evaluate new insights to new and existing products and services for the company. The difficulty lies in the implementation and realization of that particular new technology. Doing market research and competitor analysis is not enough to be ahead in times of constant innovation pressure. However, these kinds of innovation projects tend to be very risky due to high uncertainty.

There are different parties involved in transferring technology from one domain to another. When speaking of new technology development, there is always an inventor, the research and development departments of both parties, the sales departments to meet customer and market demands, the legal departments that discuss the conditions and use of the intellectual property, as well as the engineering that develops the product to serial production, and a general management that backs up the project (Barmaksiz, 2018). Another requirement is to have a stable and functioning exchange of information. The outcome as well as the workshare should be defined and clear to every party involved.

6.3 Transfer in Interdisciplinary Settings

One way to solve the uncertainty problems of innovative products and projects is to have an interdisciplinary work project on that subject. It is of great help to see into other domains and industries and see how problems are addressed differently. When there are already developed technologies and experiments, savings in technology development is a welcomed side-effect, for the fundamental research was already implemented in a real-world environment or at least further developed in applied research. With the help of the experienced domain partner, the learning curve is much shorter and therefore, potential pitfalls can easier be avoided.

Several independent research works have shown that interdisciplinary settings in research and development hold great potential for innovation and creativity. Moreover, it has shown to be effective in addressing recent global problems, such as global warming (Borge & Bröring, 2017, p. 311). Apart of that, existing solutions and proof of concepts in academia seem to be generally applicable in many industries when dismantled to a high degree of abstraction and viewed from a meta level. In some cases, technologies can be transferred almost one to one. In research centers, most often the scientist that made the finding is responsible to find applications in other domains. At the time of writing this thesis, there were confidential information existent about transfer of a physical technology from a commercial sector back to another field of applied physics. Technology is being interchanged between domains and applied in different environments and problems (Barmaksiz, 2018).

Despite the advantages, there are also some obstacles, such as different background knowledge and different language used in those specific domains. Ideally, all parties should be involved from the research stage on in the project to establish a solid knowledge base. That this cannot be the case in most cases when industry and academia work together, is unfortunately the standard. Moreover, there might be differences in interest and expectations, which should be considered by the management, as well as different standards and regulations that apply to those specific fields (ibid., p. 313). It could be of help to have a standardized form of addressing problems in interdisciplinary settings.

6.4 Methods used in Technology Transfer

When implementing technologies from institution A to institution B, regardless of being from academia or private sector, there are different methods used by the inventing scientists and engineers. This chapter presents a short summary along the operational chain of the evaluating researcher. Therefore, the used methods also outline the chronologic use in the whole process of technology transfer (see Fig. 6.3). The structure and methods follow the model of best practices in research and industry regarding technology transfer. The rough procedure is based on interviews with the innovation and technology transfer office of Germany's largest accelerator center, the Deutsches Elektronen-Synchrotron (DESY), although some methods are based on prior experience in innovation management and development.

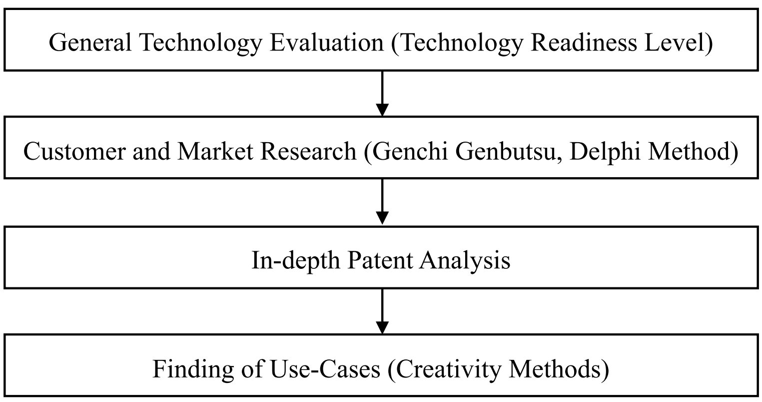

Fig. 6.3: Rough chronological order of the methodological overall approach to technology transfer

6.4.1 Technology Readiness Level

In the 1960s NASA tried to develop a method to qualitatively measure and evaluate technologies and their maturity (Hicks, et al., 2009, p. 158). This so-called Technology Readiness Level (TRL) has been the standard for the evaluation of space technologies and their degree of maturity since 1988 in accordance with ISO 16290. It is also used by the European Space Agency (ESA), the Federal Aviation Administration (FAA) and other government institutions. The degree of maturity of the investigated technology is determined on a scale (Fig. 6.4). The scale ranges from TRL 1 (observation and description of the basic principles, start of basic research) to TRL 9 (qualified system with proof of successful real-world use) (Hicks, et al., 2009, p. 159 f.).

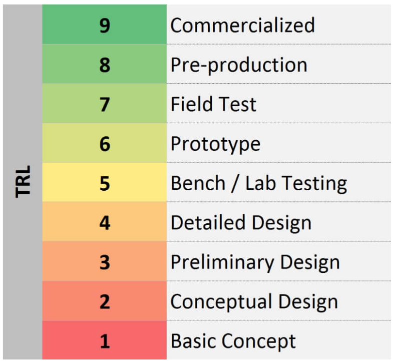

Fig. 6.4: Representation of the Technology Readiness Level on a heatmap thermometer (Dvorak, 2016)

With the help of this scaling, a technology that is still not ready for real-world use can be evaluated and used for comparison with other technologies. The use of a qualitative measurement of technology maturity levels can also support the company's internal examination of technology and the approval of resources. Combined with an extensive patent analysis, there can be potential for valuable industry insights.

However, Hicks et al. see shortcomings in some respects. Systems that require several technologies with different degrees of development cannot be mapped on the scale, as the interplay of differently weighted technologies is not evident. Furthermore, there is no use case evaluation involved in the ranking, which makes it difficult for private companies to invest in basic research (Hicks, et al., 2009, p. 160 f.).

In addition to the TRL, there are other macro -level parameters for early stage technology development regarding (Alexander, 2017), including:

- Research and Development Degree of Difficulty
- Capability Demonstrations
- Advanced Degree of Difficulty

- System Readiness Level
- Integration Readiness Levels
- Implementation Readiness Level
- Manufacturing Readiness Level
- Macro-level technology performance or complexity factors

For the purpose of having a technology development evaluation metric, the TRL is completely sufficient in the scope of this thesis.

6.4.2 Genchi Genbutsu

Genchi genbutsu is probably the main practice that fundamentally differentiates Toyota's concept from other technical management approaches. Genchi genbutsu (literal translation: *actual place, actual thing*) comes from the Japanese language and is often translated to *go and see*. This philosophy stands for visiting the place of happening in order to understand a problem or situation, both theoretically and practically (Liker & Meier, 2007) as well as from a customer, worker and employee point of view. All those who are significantly involved in development and solution of a certain problem are encouraged to have a conscious look on the matter before important decisions are made. Although following example rather deals with the last steps of experimental development, it shows the importance of senior employees identifying themselves with a specific environment or problem:

Toyota's 2004 Sienna Minivan model was to be developed primarily for the North American market. Since the chief engineer, Yuji Yokoya, who was responsible for the product did not have the necessary market expertise, he suggested a round trip throughout North America by car. The extent of the road trip with a total of 53,000 miles driven through all 50 states of the USA, 13 provinces of Canada and all parts of Mexico was enormous. Yokoya rented the current model of the Toyota Sienna in small and large cities and drove a certain distance together with potential customers. He observed them and recorded first-hand information about what potential customers from North America would expect from a minivan (Ries, 2015). Yokoya collected and measured high-quality data directly from the customer, from which he was able to draw conclusions of a huge quantity. One of the most important findings was that although the customer is the buyer, the users can certainly influence this purchase. With a minivan, these were usually the customer's children or grandchildren. For this reason, Yokoya understood that the new model had to be made more appealing to children. More focus should be placed on comfort, as longer distances are usually covered in North America than in Japan (Ries, 2015, p. 82 ff.). Without this method, the product would be developed in a wrong direction. It helps to prove hypotheses and test it in a real-world environment.

In technology transfer, this approach helps to understand the domain specific situation. It is of immense help that a researcher goes to the actual place and sees for himself. An extensive analysis of the real-world situation and environment is inevitable when trying to implement a technology in a rather unknown industry. Since the main orientation of private companies are their customers, the needs of the customers are important to understand, as well, when wanting wo transfer technology to the private sector.

6.4.3 Delphi Method

Named after the ancient Greek oracle, the Delphi method is a technique to obtain a collective view from individuals about issues where there is no or little definite evidence and where opinion is important. Initially, this method comes was used by the military to predict probable effects of massive atomic bombs. The objective of the original study at Rand Corporation was to *obtain the most reliable consensus of opinion of a group of experts ... by a series of intensive questionnaires interspersed with controlled opinion feedback* (Linstone & Turoff, 2002). Now, this technique is used in many environments, including economic and financial settings, technology and trend analysis, and healthcare. (Thangaratinam & Redman, 2005). As a whole, the Delphi process is characterized as a method for structuring a group communication process with the aim to allow a group of individuals to deal with a complex problem. Four things are provided to accomplish this structured communication: feedback of individual contributions of information and knowledge, assessment of the group view, opportunity for individuals to revise certain views, and a degree of anonymity for the individual responses (Okoli & Pawlowski, 2004).

In both, the classical method of surveys and the Delphi method, a questionnaire is designed and presented to a group of individuals that answer the questions based on the personal knowledge about the matter. The questions that a Delphi study investigates are of high uncertainty and speculation, thus not appropriate for a general population. In a Delphi study the group of experts, about 10 to 18 is handpicked by the researcher based on the qualification of the experts. After the survey is done, the researcher then analyzes the responses. A group feedback helps to prioritize the next round's questions. In addition to new questions derived from the group feedback, original questions can remain for experts to revise their response. This process is an iteration loop until the experts reach a satisfactory degree of consensus (Okoli & Pawlowski, 2004).

The process is split in roughly two steps, selecting experts (1), and the administration of the surveys (2), each step consisting of subsets. The actions of each step are described in the following table (Table 6.1).

Table 6.1: Overview of the Delphi method's steps (Okoli & Pawlowski, 2004)

Step	Description of Tasks
1.1 Identification of relevant sets	Identify relevant disciplines or skills in academics, practicioners, NGO and government officials, Identify relevant organizations, and academic literature
1.2 Identification of experts	Identify experts of above named disciplines, skills, organizations, academic literatur
1.3 Nomination of additional experts	Contact experts listed in 1.2 and ask for further nomination of other experts
1.4 Rank experts	Categorize experts in different sub-lists or panels, one for each discipline, Rank experts of each list based on qualification
1.5 Invite experts	Invite experts in order of their ranking, within their discipline sub-set, Stop when target size of 10 to 18 experts is reached for each panel
2.1 Brainstorming	Ask experts as individuals to assess Questionnaire 1 (for example listing relevant factors for an optimal infrastructure for a digital factory), Consolidate the lists of all experts, remove duplicates, and unify terminology, Send Questionnaire 2 (consolidated list) for validation and refine final version
2.2 Narrowing down	Send Questionnaire 3 (finalized version) to each expert panels and ask to select at least ten factors, For each panel, keep factors which were chosen by over 50 percent of experts, use those for the next questionnair
2.3 Ranking	Questionnaire 4: Ask experts of each panel to rank factors, Calculate mean rank for each item (based on all panels), Get feedback of every panel, share feedback, and ask to re-rank the list, Reiterate until experts reach consensus

In research, the Delphi method is especially useful to identify research topics, specify research questions, identify a theoretical perspective for the research, selection of variables of interests and generation of propositions, preliminary identification of causal relationships and definition of constructs and the creation of a common language for discourse (Okoli & Pawlowski, 2004).

6.4.4 Patent Analysis

A patent analysis answers many questions when investigating a technology transfer. Thus, it is a key method to understand the transfer situation. Due to patents being available online and browsable in real-time, it is possible to filter and specifically search for keywords. This answers which patents were filed by whom, which countries are specialized in which technology, which collaborations take place in which industry, and which domain filed how many patents regarding a specific technology. A patent search and visualization tool is helpful in finding the right information through all the patents, since with about 50 million internationally filed patents from 1985 to 2016 patent research can be time-consuming (WIPO, 2018).

Since patents contain extractable knowledge, visual tools are crucial. To visualize the information, machine learning and data science methods come into play. Below figure shows a framework of a visualization method for patent analysis proposed by Kim, et al. (Fig. 6.5).

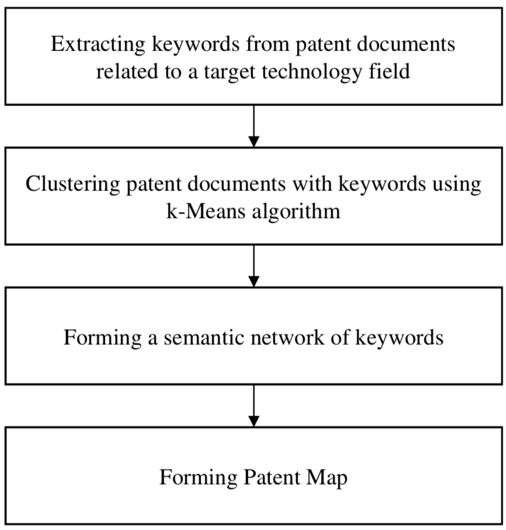

Fig. 6.5: Steps of a framework to visualize patent information (Kim, et al., 2008)

A detailed explanation of every step would be out of scope of this chapter; therefore, an example of patent visualization is given below (Fig. 6.6).

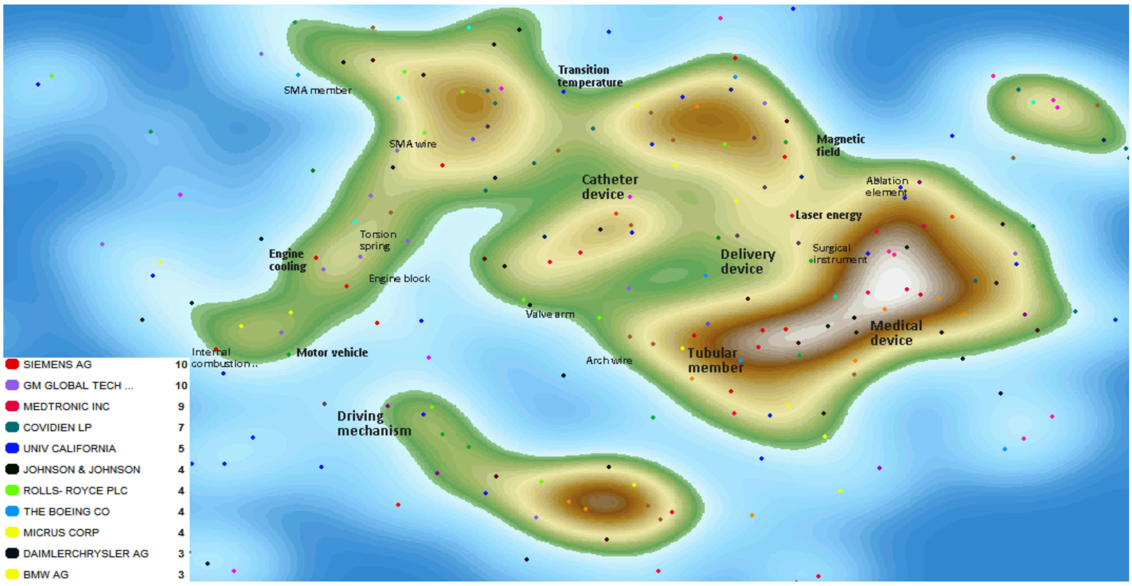

Fig. 6.6: Multi-dimensional cluster map to visualize patent keywords and institutions that submit them; Mountains represent high number of patents filed (Gridlogics, kein Datum)

From the tools that were investigated in the research of this chapter, all were focused on an intuitive user interface with user interaction and vast filter possibilities to further ease the knowledge extraction. Of those tools were PatBase (commercial tool), DEPATISnet (Germany), Espacenet (Europe), PATENTSCOPE of WIPO (International).

At the time of writing, accessing PATENTSCOPE for the visual inspection use case in Chapter 7, the database brought forward the following number of patents (Table 6.2).

Table 6.2: Findings in the patent database of WIPO regarding drone-assisted visual inspection keywords

Keyword	**# of patents filed**
Visual Inspection	201,415
Machine Learning	73,086
Deep Learning	5,673
Computer Vision	45,373
Unmanned Aerial Vehicle / UAV	28,948
UAV ∧ intelligent	2,150
Inspection ∧ UAV	1,835
(Inspection ∨ Computer Vision ∨ Deep Learning) ∧ UAV	2,429

Out of around 69 million worldwide patents available online, 1,957 patents deal with an unmanned aerial vehicle paired with intelligent functions like computer vision or deep learning, or for the purpose of inspection. The patent texts are accessible, along with the information on who and which company filed the patent. Due to non-accessibility of other visual tools for patent analysis, a further research is not possible at this point in time.

6.4.5 Creativity Methods

Out of the gathered information, specific use cases can be found or built with the help of creativity methods. There are a lot of creative methods for problem analysis, and idea and solution finding, the most prominent one being brainstorming. Since the further explanation of creativity techniques would be out of scope of this thesis and this chapter, the general method is being explained, with the addition of a list of helpful techniques.

The key in finding solutions for a problem (or possible answers to a question) is a structured path when using creativity techniques. Sinfield, et al., proposed a seven-step process, based on idea generation approaches from different backgrounds and domains (Fig. 6.7).

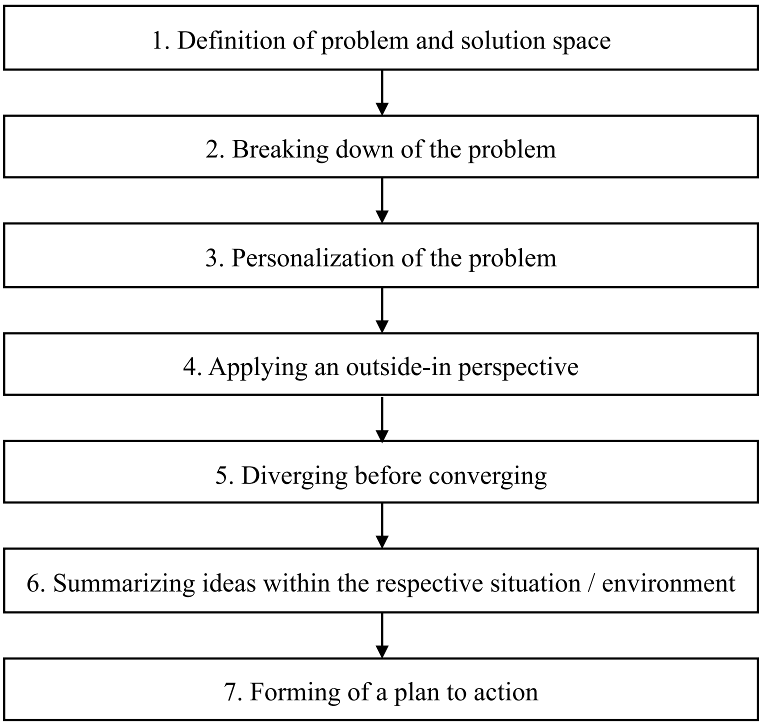

Fig. 6.7: Outline of a structured problem solution proposal; Own representation based on (Sinfield, et al., 2014)

While the first three steps are designed to help the participants understand the problem profoundly, steps 1 to 6 are about idea generation, before they are formed into incisive action plans. Especially the fifth step, the essence of idea generation (divergence and convergence), helps broaden the view to gather as much ideas as possible and then narrowing it down to realistic and most-promising solutions (Sinfield, et al., 2014). Solving steps 1 to 4 with the methods of this subchapter, such as the Delphi method or genchi genbutsu, is imaginable. Acknowledging the diverging before converging premise, the use of creativity methods gets easier and with a more accurate aim at generating new ideas. Below table represents different creativity techniques for the two different creativity phases (Table 6.3).

Table 6.3: Creativity methods divided in divergent and convergent ones (Vullings, 2013)

Divergence	**Convergence**
Attribute Listing	COCD-Box
Biomimicry	Enhancement Checklist
Brainwriting (Method 3-6-5)	Force-Field Analysis
Challenge Assumptions	Hundred Euro Test
Osborn Checklist	Idea Advocate
Classical Brainstorming	Negative Selection
Excursion Technique	NUF Test
Harvey Cards	PINC Filter
Imaginary Brainstorming	Six Thinking Hats
Lotus Blossom Technique	Weighted Selection
More Inspiration	
Personal Analogy	
Random Input	
Redefinition	
Reverse Brainstorming	
Systematic Inventive Thinking	
TRIZ	
Wishing	

Due being out of scope, a further explanation of the creativity methods is not given.

6.5 Similarities in Chapter 5's Use Cases

With regard to Chapter 5 and the use cases described, to further analyze how to gain benefits from the applications and use cases in medicine to aviation, a more cost-effective approach which is used in technology transfers, is being proposed for this thesis' setting. Instead of doing basic research or develop industry-specific technology from the outset, parallels between use cases in other industries are ought to be found. When talking about innovative technology and product development, the high risk is a key factor that raises the costs, and therefore discourages the management to further invest in technology research.

When it comes to applications of artificial intelligence and deep learning, there are two possible ways of a simplified transfer. The applied AI technology, like computer vision, gets implemented in aviation directly, or through a medical funnel using the learning curve of the already solved problems and overcame difficulties in medicine (Fig. 6.8).

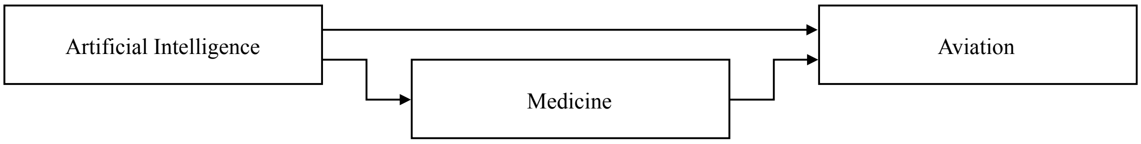

Fig. 6.8: Technology transfer possibilities from AI into aviation

Since this thesis deals with the planning and implementing of deep learning use cases into an aviation environment from the learnt experience of the medical industry, the second path of above figure applies here. And since the use cases in Chapter 5 were chosen without any glance of possible similarities beforehand, the knowledge used to solve problems of those specific use cases should show the general possibility of using another domains problem solving approach (Figs. 6.9 and 6.10).

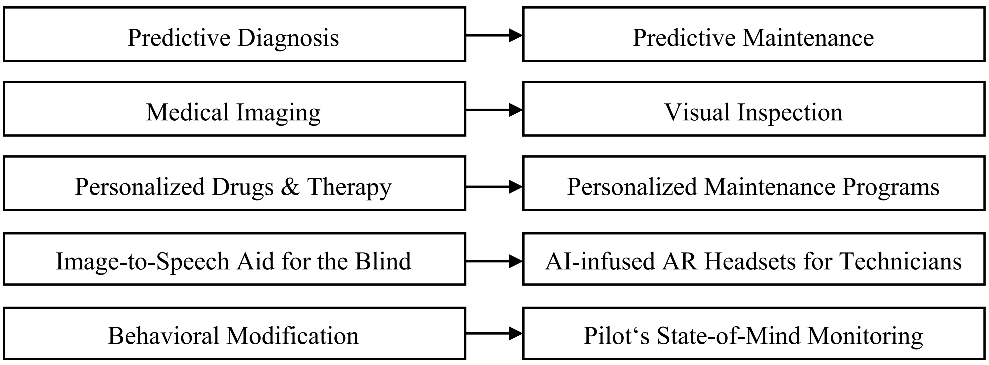

Fig. 6.9: Product concepts and use cases for aviation derived from Chapter 5's medical use cases

The use case of predictive diagnosis, as described in 5.1.1 uses multilayer perceptrons (Choi, et al., 2017) to classify and predict heart failures on data basis consisting various parameters. The case with predictive maintenance is similar: If there is a sufficient data basis, classification of potential harms and errors can be made with multilayer perceptrons. In the case of time-series events, recurrent neural networks are the better choice (see subchapter 4.1.3).

The similarities between medical imaging and visual inspection are going to be explained in detail in Chapter 7.

Personalized maintenance programs can be extrapolated from personalized drugs and therapy. Yet, the data basis is different since for personalized drugs the possible genomic links to diseases were already in most cases analyzed in medical research, and in personalized maintenance programs have to be preprocessed. Either way, in both cases generative models can be used to simulate specific situations, e.g. simulate new drugs or simulate aircraft operation and routes.

The image-to-speech aid for the blind use case is very product-driven, as is the aviation counterpart, where the exact same technology architecture can be used to help maintenance and repair technicians with additional information; be it as audio or video output. Although the product was not explained in detail, the used frameworks in both cases would be a video signal as input, processing it through CNN's (see subchapter 4.1.2) to detect objects, enhanced with an RNN with a specific language model to give a descriptive audio output, as well as other functions such as speech recognition, if the blind person or the technician respectively, would want to interact with the system.

The last medical use case is about sensor input data and classifying, or predicting respectively, certain move patterns to detect smoking habits with the help of LSTM networks (see subchapter 4.1.3). In aviation, sensors could also monitor the pilot's state of mind and predict the next moves or detect irregularities in handling the aircraft as well as irregularities of the aircraft and aircraft components itself.

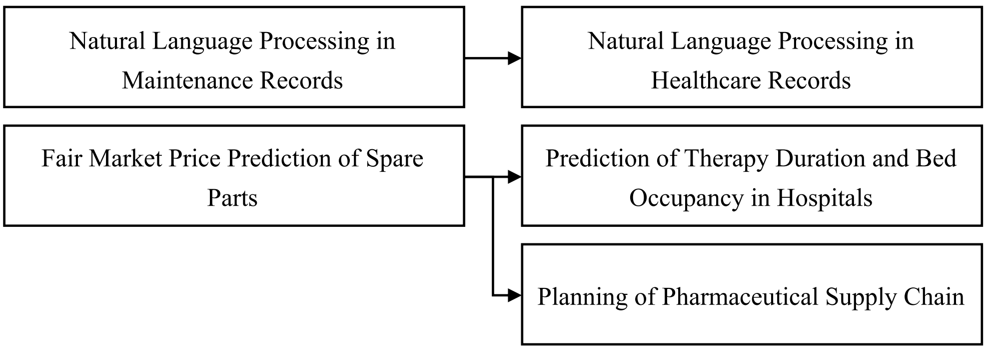

Fig. 6.10: Use cases for medicine derived from Chapter 5's aviation use cases

When speaking of handwritten maintenance records, multilayer perceptrons would analyze the written reports for character recognition and an RNN and a natural language model would map it to a generalized language. The exact same could be done with healthcare records that are also handwritten and digitized manually.

Last but not least, the fair market price prediction is based on reinforcement learning, which itself is a powerful method to analyze and find own patterns in a specific task (see subchapter 4.1.5). Used in teaching AI how to play video games, every kind of simulation or business game could profit from how an AI solves problems. It has to be understood, that humans can be beat in such a game, which makes it even more promising in factory layout and production simulation in aviation. In healthcare, a hospital can simulate better alternatives to the existing bed and staff planning. It is also imaginable that the supply chain, be it in medicine or aviation, can be further optimized with such a simulation game.

Of course, the methodological approach explained in subchapter 6.3 should be followed, to extract as much as possible knowledge from the analyzed use cases and find other domains and possibilities of adaption. This chapter however, only discusses the similarities between Chapter 5's use cases and only takes into account the situation in the medical as well as the aviation industry. A more applied approach is given in Chapter 7, where a technology that comes from computer science and artificial intelligence is being used in medicine and could also be adapted to aviation. In addition, the description of the use case is problem-driven to fully grasp the situation. Doing this for every use case in Chapter 5 would be impossible considering the boundaries of this thesis.

7 AutoInspect

Damages on aircrafts are of various kind, including bird strike, stone strike, lightning strike as well as damages from other airport vehicles, which is called aircraft ground damage. In some cases, these damages go unreported. Although the major number are minor issues like dents or scuffs, undetected structural damages can carry very serious damaging potential. With the increasing use of composites, invisible damage can be one of the more serious outcomes of it (Pierobon, 2015). Of the more severe damages are also bird strike, of which the most struck parts of the aircraft are wind shields, engines, radomes and wings (Hedayati & Sadighi, 2015, p. 9 ff.).

As it is the case with computer vision, deep learning frameworks like CNN's have shown to outperform humans in general visual recognition tasks (He, et al., 2015). Matched with powerful detection algorithms like YOLO, this becomes a strong tool for industrial use. Visual detection of damages is therefore a promising use case in aviation, and in general large and expensive objects, especially if it is linked to serious safety issues. This chapter deals mainly with damages by lightning strike. A detailed transfer possibility is also given in this chapter, mainly with findings in genomic detection in medicine. AutoInspect therefore refers to a possible implementation of damage detection in the maintenance of aircraft.

7.1 Damage Protocol

Safety authorities in aviation dictate a defined procedure when an aircraft gets hit by a lightning strike. An aircraft operator is legally bound to check the aircraft's skin for damages right after landing and is not allowed to carry on with further commercial flights. The maintenance division of an aircraft MRO shop is usually split into types of aircraft or routes: short haul, long haul and cargo. Whenever there is the need to further investigate damages, the maintenance technicians contact the engineering department and pass on the description and photographs of the damage for consultation. Depending on the damage, there are engineering experts in structure, engines, and other parts of the aircraft, that define the next steps. Besides lightning strikes, there are several other types of damages, including dents, scratches, holes, abrasion of the engine cowling, and scrubbing due to vibration and drag. When talking of lightning strikes, there are three types of strikes: relating the skin, the fasteners, or both. Below figures show examples of the types lightning strikes on different aircrafts (Figs. 7.1, 7.2 and 7.3). An aircraft can get hit a few times up to 50 in one thunderstorm. The lightning strikes can leave burn marks behind that come most often with a characteristic yellowish stain around the damaged area. Damages can occur in a range of 1 mm up to more severe structural damages (Rindt, 2018).

Fig. 7.1: Lightning strike to the skin with a burn mark (LHT, 2018)

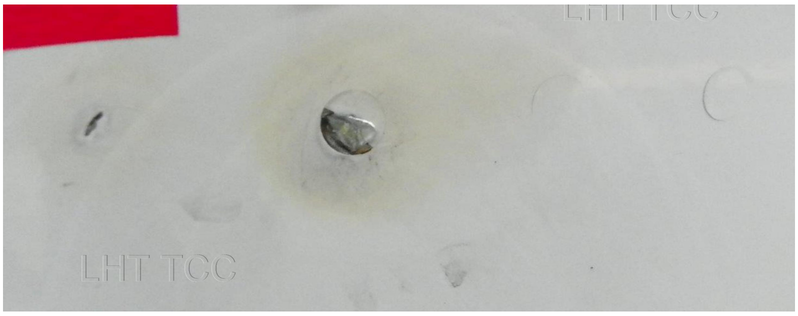

Fig. 7.2: Two fasteners hit by lightning strike; other two undamaged (LHT, 2018)

When such damages are found, after classification whether the damage is minor or major, further steps are to be considered to attest the health and airworthiness of the aircraft, including measuring residual thickness of the damaged part of the skin and repairing the metal sheet. The repair work, including grinding and polishing the affected area, is in accordance to the Structure Repair Manual (SRM). Due to the melting and remove of material after a lightning strike, the static load capacity can decrease. A residual thickness measurement ensures structural intactness when below limit, and a compulsory attachment of additional metal sheet onto the aircraft's skin. Sometimes, lightning strikes occur near important instruments, such as static ports that measure the static air pressure. An investigation makes sure the damage is within the limits and not too close, otherwise the pressure instrument has to be replaced precautionary (Rindt, 2018).

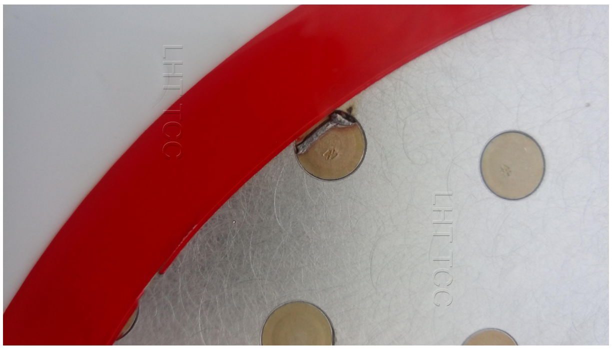

Fig. 7.3: Damage inside the static port area; need for further investigation (LHT, 2018)

The repair of the damage is then documented, most often in handwritten protocols, and stored (Rindt, 2018). In the Appendix, there is one image given each for a hard to classify type of damage or probably false positive result respectively, because of old paint or dirt on the aircraft skin (Figs. A2 and A3).

7.2 Problems in Manual Inspection

Along the process chain, from registering of the lightning strike by the flight crew to the repair of the damage, there are a lot of problems. Since the flight crew has no digital reporting opportunity, damages are registered in a handwritten protocol. Although the handwriting is clear and easy to understand, there is no standardization. However, for the purpose of notification this is sufficient. The main problem lies in the detection of damages. Despite the fact that maintenance technicians are experts in finding damages caused by lightning strikes, the circumstances are not standardized. The technicians sometimes have no access to a hangar, thus have to inspect the aircrafts on the apron of the airport. This comes with the changing of weather conditions or inspections at night equipped with a flashlight that can make detection difficult. Nonetheless, during the interviews conducted, there were no signs of missing a damage due to irregularities in the maintenance environment. The search process varies in time, taking up to several hours in many cases. Understandable, because the technician has to inspect every inch of the aircraft's skin from about 1.5 meters distance which constitutes another problem: human fatigue (Grönheim, 2018). Fatigue in high-concentration work is a problem that can bring down efficiency and quality of work. The technician has to have a balanced work schedule to be able to maintain the high concentration needed for detecting damages. This is similar to the fatigue problem in visual quality recognition discussed in subchapter 5.2.2.

When damages are found, the technician marks the area with a permanent marker. To further make things easier for the repair, the place of damage needs to be classified too, in terms of aircraft structure. This includes the number of section of the aircraft's structure and fuselage and is done so other technicians do not have to search intensively for the damage. Although seldom, it can also happen that no damages are found. In this case the damage report looks accordingly. Classification of the damage finds place when they are found, sometimes with the consultation of an expert maintenance engineer. Additionally, a photograph of the damage is taken. This process represents a problem for the use case since markings on the aircraft's skin make it difficult for a machine to get trained to detect damages. Since the classification task, whether by a human or a machine, is done on a clean aircraft, it is not correct to train a machine with photographs that contain markings. This results in a lot of preprocessing of the training data, including cropping, editing, setting hue and saturation, removing color channels of the marking color, and on the other hand enhancing certain color channels, e.g. the yellowish burning marks. The oldest photographs of damages that were existent at the time of writing this thesis and searched through the LHT damage database were around 10 years old. Some older photographs were not usable due to bad resolution and image quality (Rindt, 2018).

7.3 Algorithmic Detection of Damages

Detection of damages are similar to other object detection tasks. Represented as a simple black box, an object detection algorithm would look like the figure below (Fig. 7.4). The relevant frameworks are presented in 7.4.

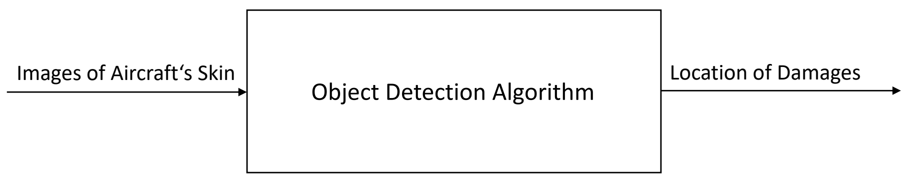

Fig. 7.4: Simple black box to show input and output of an object detection algorithm

Detached from the actual approximation in a neural network (with features from low to high level), from a heuristic point of view, a detection problem would be decomposed in sub-problems. When recognizing a face, trivial questions to fulfill the main task might be if there is an eye in the top left, an eye in the top right, a nose in the middle, etc. In the case of a lightning strike damage, the questions a technician subconsciously would answer to come to a conclusion are represented in Fig. 7.5.

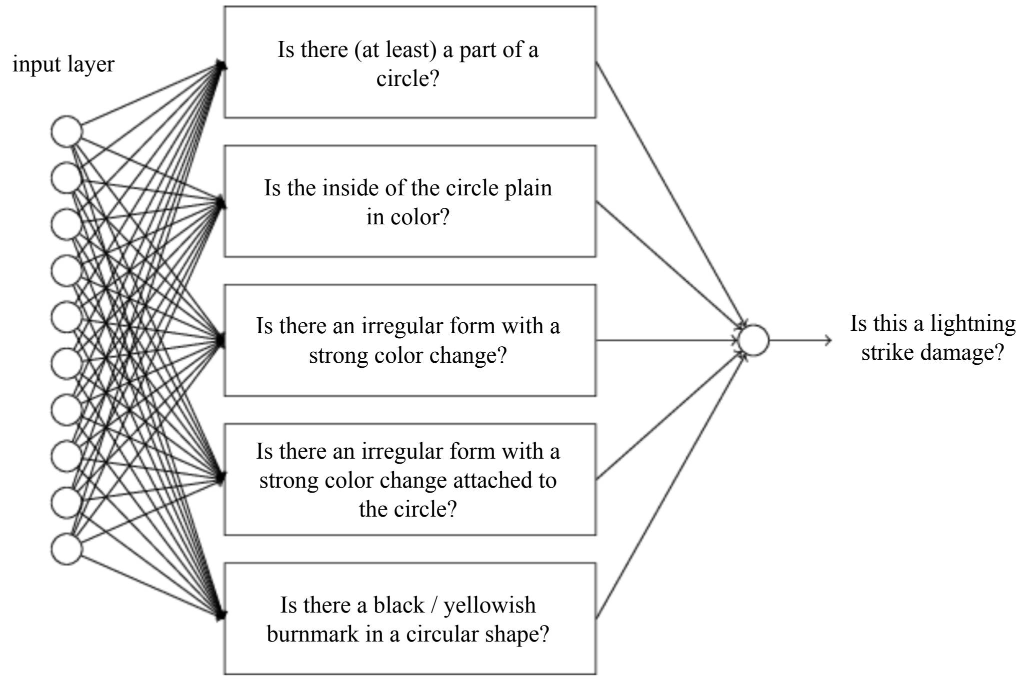

Fig. 7.5: Intuitive approach of how neural networks would work in lightning strike damage recognition; Representation based on (Nielsen, 2017)

Yet, detection and recognition are not the same. To get a better understanding about the task, a definition of *detection* is needed at this point. Object detection, unlike mere classification or recognition, includes the information of *where* a certain object or class is in an image (Huttenlocher, 2009). This is especially important for detecting damages, whether done by a machine or a human, since more than one technician works on an aircraft and the work is split into several subtasks. It is therefore crucial that the Information about the location of the damage is registered. From an abstract point of view, an image is built of three basic information: object, position and orientation (Bishop, 2009, p. 366). The object detection algorithm would take the information of the aircraft's skin in form of an image as an input and would output whether there are damages or not, and most importantly where exactly they are located.

Below figure shows the difference between mere classification and detection. Another step further, explained in detail in subchapter 7.3.4, is instance segmentation (Fig. 7.6).

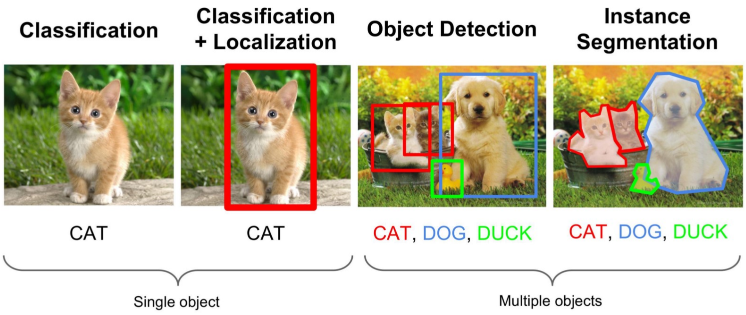

Fig. 7.6: Difference between image classification, object detection and instance segmentation (Ouaknine, 2018)

What this chapter does not include are the different datasets since the task of damage detection is very specific. Several datasets were released in the last years, including the 2012 PASCAL VOC dataset, the 2014 ImageNet dataset, and the 2015 COCO dataset (Russakovsky, et al., 2015; Lin, et al., 2015). A description of how the dataset should look like in the case of damage detection is given in 7.3.2.

However, what the analyzed object detection algorithms have in common are same performance metrics. Of those are Intersection over Union (IoU) and mean Average Precision (mAP). In the case of binary object classification, the task is to just output whether the object is in the image or not. For object detection, this is not sufficient. In intersection over union (see Fig. 7.7), the ground-truth (e.g. manually set) bounding box of the object and the predicted bounding box are overlapped. The higher the IoU value, the better the predicted location of the object's bounding box. Detections with an IoU value lower than a set threshold will not be further considered (Ouaknine, 2018). In most cases, an IoU value > 0.5 is considered as a good prediction (Rosebrock, 2016).

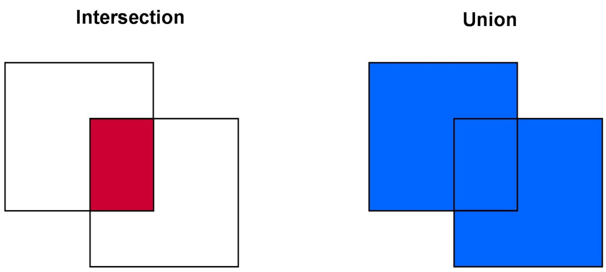

Fig. 7.7: Visualization of the intersection over union-method (Rieke, 2017)

The mean average precision is based on precision, which can be explained as how many detected objects are relevant (or false positive rate), and recall, which is about how many relevant objects are detected (or false negative rate). The average precision is calculated by taking the average value of the precision across all recall values. Having multiple classes, the mean average precision rate comes into play, which takes the mean of average precisions over all classes and IoU tresholds (Arlen, 2018).

7.3.1 Preprocessing

Of the more time-consuming tasks is the preprocessing of the damage images. Since a digital image consists of RGB values for each pixel, what a machine sees is different than what a human perceives with the naked eye. The image has to be brought in the right setting with the right parameters, such as cropping the damage region, downsizing, gray scaling, and adjusting hue and saturation.

The preprocessing takes place before the images go into the neural network. This important step ensures that the network only gets the relevant information. The preprocessing in this case is done with the original image brought down in size to 25 by 25 pixels (Fig. 7.8).

Fig. 7.8: Image of a damaged fastener, cropped and downsized to 25 by 25 pixels; from (LHT, 2018)

A code written in Matlab (see Appendix) gets the size of the image and assigns an 8-bit integer for the pixel intensity (gray scale) for the 25 by 25 value map. A loop then adjusts the intensity for each of the RGB channels according to preset parameters. Before the processed images are output, the output range of intensity values are adjusted to get a richer contrast. The contrast adjustment is done twice, to get a comparison how much of it is enough to get the relevant features extracted. The result of the preprocessing is presented in the figure below (Fig. 7.9).

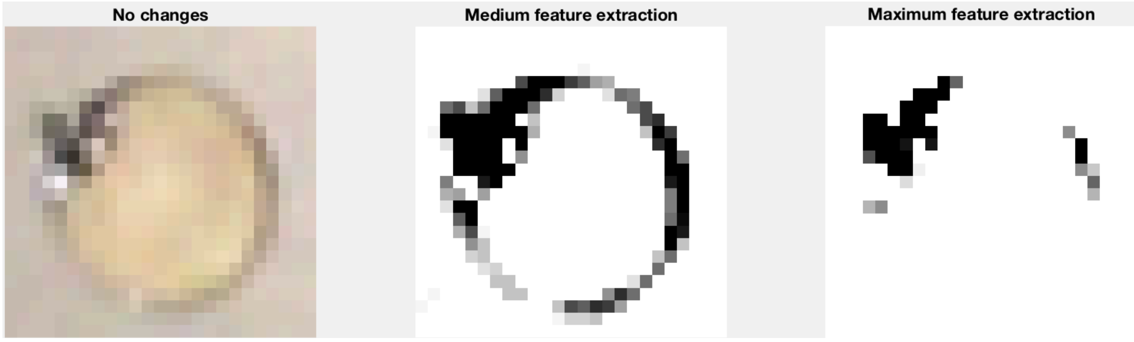

Fig. 7.9: Result of preprocessing the damage image with two levels of intensity value range adjustment (center and right); The unprocessed image is for comparison (left)

The output 25 by 25 intensity value matrix *intval* is what a machine *sees* as its perceiving its environment, in this case a lightning strike-damaged fastener. Attaching it to a gray scale or intensity color map (see source code in Appendix), the matrix makes sense visually (for humans) and represents the processed input for the neural network (Fig. 7.10).

255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	0	0	0	0	45	95	162	112	212	255	255	255	255	255	255	255	255
255	255	255	255	255	245	78	0	0	0	0	212	228	255	255	228	112	162	255	255	255	255	255	255	255
255	255	95	28	95	162	0	0	0	78	255	255	255	255	255	255	255	95	28	255	255	255	255	255	255
255	255	0	0	0	0	0	0	245	212	255	255	255	255	255	255	255	255	45	45	255	255	255	255	255
255	228	12	0	0	0	0	0	0	178	255	255	255	255	255	255	255	255	245	0	0	255	255	255	255
245	255	212	0	0	0	0	255	0	255	255	255	255	255	255	255	255	245	255	0	0	255	255	255	255
255	255	255	0	0	0	0	228	95	255	255	255	255	255	255	255	255	255	255	228	0	95	255	255	255
255	255	255	28	0	0	0	0	255	255	255	255	255	255	255	255	255	255	255	255	0	0	255	255	255
255	255	162	255	255	28	0	255	255	255	255	255	255	255	255	255	255	255	255	255	28	0	255	255	255
255	255	78	228	255	162	255	255	255	255	255	255	255	255	255	255	255	255	255	255	178	0	255	255	255
255	255	228	0	0	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	145	0	255	255	255
255	255	245	45	0	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	28	28	255	255	255
255	255	255	178	78	178	255	255	255	255	255	255	255	255	255	255	255	255	255	255	0	28	255	255	255
255	255	255	228	112	212	255	255	255	255	255	255	255	255	255	255	255	255	255	255	0	145	255	255	255
255	255	255	255	178	212	255	255	255	255	255	255	255	255	255	255	255	255	255	212	95	255	255	255	255
255	255	255	255	228	228	255	255	255	255	255	255	255	255	255	255	255	255	255	162	255	255	255	255	255
255	255	255	255	255	245	228	178	245	255	255	255	255	255	255	255	255	228	95	228	255	255	255	255	255
255	245	255	255	255	255	255	212	162	255	255	255	255	255	255	162	12	145	228	255	255	255	255	255	255
228	255	255	255	255	255	255	255	228	255	255	212	28	45	12	45	178	255	255	255	255	255	255	255	255
228	255	255	255	255	255	255	255	255	255	255	245	212	228	178	255	255	255	255	255	255	255	255	255	255
255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255	255

Fig. 7.10: Assignment of a gray scale color map to the intensity value matrix

In this specific case, there is no need for the RGB color channels, as the features are better visible and distinguishable from the rest of the skin in a gray scale. In cases where many classes have to be classified and detected, e.g. real-time detection based on color features such as tomato sorting or autonomous vehicles, RGB channels would make more sense as there is a greater need for the containing information.

7.3.2 Boundaries and Further Preparation

Visiting the maintenance branch of Lufthansa Technik in Frankfurt, an access to the damage database was possible. The database included several thousands of damage images and protocols, and more than one thousand lightning strike damages. Due to restricted accessibility, the correct number is not known. About 550 images were handpicked and extracted. Although the number is small, almost all images have to be extensively prepared and preprocessed to meet the requirements of a training dataset. The small quantity of damage images makes it hard to meet quality requirements in aviation since the learning curve rises and simultaneously the error rate drops with more samples. At this point already, it has to be said that the small number of images is a serious constriction. Furthermore, the image quality as well as the homogeneity, environment settings such as lighting, indoors or outdoors, and weather conditions vary immensely. That said, for a deep learning network, the variance is not an obstacle naturally, if there are enough samples. This means for every kind of damage, having a dataset of 1000 images would be sufficient, and additionally at least the same number of undamaged images, referring to subchapter 5.2.2. To achieve a steady quality level for the training datasets, especially when feeding more samples into the neural network for continuous learning (Parisi, et al., 2018), a standardized process regarding taking photographs of damages is ought to be considered.

Although this thesis only focusses on the transfer possibilities and the underlying theory of deep learning, more in-depth aspects such as hyperparameter optimization, e.g. number of hidden layers and units, dropout value, learning rate, and batch size, that also play a role in solving an algorithmic damage detection task, is not covered (Radhakrishnan, 2017b).

Below figure shows a possible outcome of an object detection algorithm like described in subchapter 7.3.3 and 7.3.4 (Fig. 7.11).

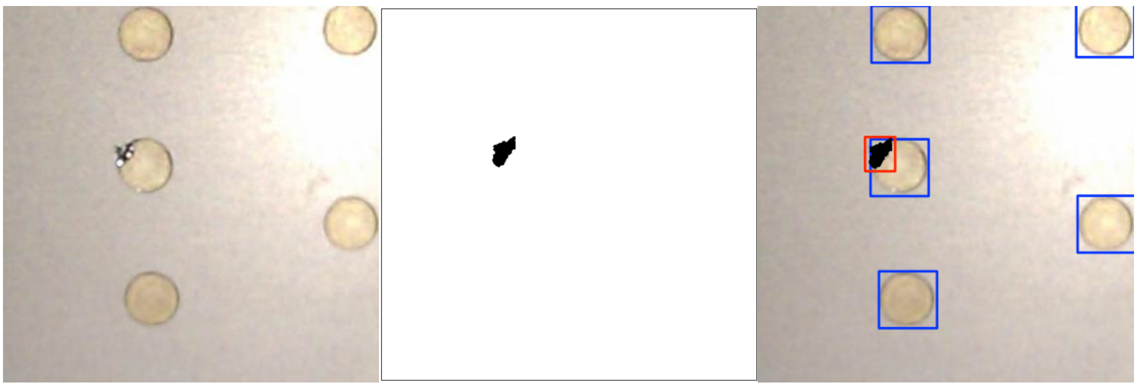

Fig. 7.11: Representation of the detection of intact features (blue) and damages (red) with the help of a damage mask (center)

Having Fig. 7.10 in mind, it is possible to detect the round circle feature (matching it with $p > 95$ percent) and thus classifying damage-free and damaged fasteners.

Since the state of the art approaches in object detection are based on CNN's (see next subchapters), the neural network framework in this case would look like the ones presented in subchapter 4.1.2.

7.3.3 YOLO

Having the fast and accurate human visual system as an example (You Only Look Once), a research team of the University of Washington and Facebook AI Research have published a paper on a simple real-time object detection algorithm. YOLO promises a simple architecture, e.g. unlike the R-CNN approach (explained in subchapter 7.3.4), where the object detection task is reframed as a single regression problem. Referring to the below figure, a single convolutional network a single convolutional neural network simultaneously predicts multiple bounding boxes and class probabilities for those boxes (Fig. 7.12). The focus in this and the following subchapter is the object detection, so the system architecture is left out knowingly.

First, the input image gets divided into an $S \times S$ grid where the grid cell that includes the center of the object is responsible for detecting that object. Each grid cell then predicts B bounding boxes and confidence scores for the bounding boxes, which is about how confident the model is that the box contains an object and how accurate the box is that it predicts. The confidence prediction represents the IoU between the predicted boxes and the ground truth box. Additionally, each grid cell also predicts C conditional class probabilities, e.g. how probable it is that a class is inside the viewed grid cell. The predictions are encoded in an $S \times S \times (B * 5 + C)$ tensor (Redmon, et al., 2015).

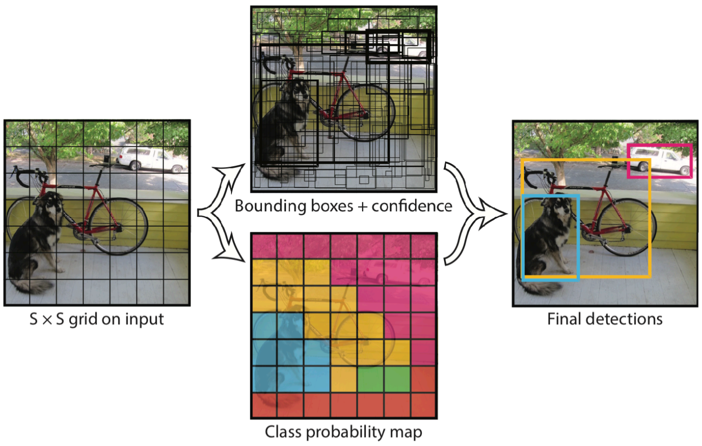

Fig. 7.12: System model of YOLO object detection algorithm (Redmon, et al., 2015)

However, since each grid cell only predicts two boxes and can only have one class, this limits the number of nearby objects that the model can predict. Small objects, that appear in groups is an obstacle. Also, it struggles to detect objects in new or unusual aspect ratios or configurations, and additionally, the set loss function treats errors the same in small and large bounding boxes. The main source of error is incorrect localization, since a small error in a small box has a much greater effect on IoU than a small error in a large box.

The YOLO algorithm reaches 45 fps on a Titan X GPU with a latency of 25 ms, while a faster version of YOLO, with a bit less accuracy, runs at 155 fps. The error analysis shows, that about 65.5 percent is correct and about 19 percent is due to localization error. Compared to Fast R-CNN, which will be explained further in the next subchapter, the error is greater (Fast R-CNN 8.6 percent), yet the background error rate with 4.75 percent is much smaller then compared with the Fast R-CNN algorithm (13.6 percent).

Since the development of YOLO, further adjustments were done, the latest being YOLOv3. What is not explained in this subchapter is the anchor box concept from YOLOv2 (also called YOLO9000) on, that on basis of ground truth labels, a predetermined set of boxes are applied, instead of directly predicting a bounding box, as well as multiple detections for each grid cell. On the PASCAL VOC 2007 and 2012 dataset, YOLOv2 performed better than Fast R-CNN's (78.6 mAP / 40 fps and 77.8 mAP / 59 fps vs. 70.0 mAP / 0.5 fps) (Redmon & Farhadi, 2016).

7.3.4 Mask R-CNN

To get to the specific concept of Mask R-CNN, a basic explanation of R-CNN's has to be given first since Mask R-CNN is based on R-CNN, as the name already suggests. Originally, CNN's begin with the region search and then perform classification (see subchapter 4.1.2). In R-CNN's, the selective search method that was developed and proposed by Uijlings, et al., is being used. Selective search initializes small regions in an image and merges them with hierarchical grouping. The detected regions are merged according to a variety of color spaces, such as light intensity and shadowing, and similarity metrics, such as texture histograms (Uijlings, et al., 2012). For a better clarification, see below figure (Fig. 7.13). Before merging similar regions, there were a lot of bounding boxes for each single region.

Fig. 7.13: Example of the selective search method (Uijlings, et al., 2012)

While for R-CNN's each selective searched and detected segment then act as a resized input for the CNN (hence the name Region-based CNN), the Fast R-CNN algorithm inputs the whole image into the CNN and detects Regions of Interest (RoI) afterwards with the selective search method on the produced feature maps. Each RoI pooling layer feeds fully connected layers, creating a feature vector, which is then used to predict the observed object and adapt bounding box localization (Ouaknine, 2018). The Faster R-CNN framework was proposed using a Regional Proposal Network instead of selective search image segmentation, where the above explained Fast R-CNN algorithm then classifies on those regions (Magruder, 2018a).

The Mask R-CNN algorithm adds a parallel branch to the bounding box detection for predicting the object's mask. The RoI pooling layer gets replaced by a new proposed method called RoI Align to calculate exact values of the object's location. This is done by using bilinear interpolation from the four nearby grid points on the feature map (Fig. 7.14).

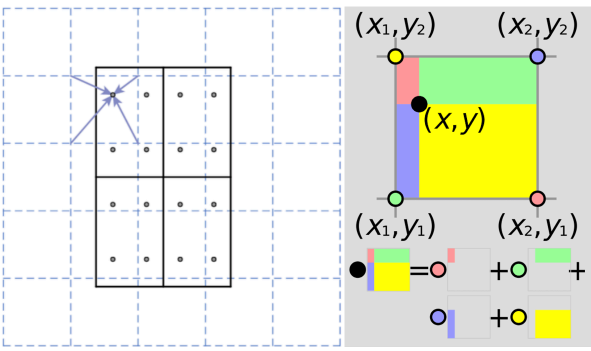

Fig. 7.14: Visual representation of the bilinear interpolation operation (He, et al., 2018; Magruder, 2018)

$$f(x, y1) \approx \frac{x2 - x}{x2 - x1} f(Q_{11}) + \frac{x - x1}{x2 - x1} f(Q_{21}) \tag{2}$$

$$f(x, y2) \approx \frac{x2 - x}{x2 - x1} f(Q_{12}) + \frac{x - x1}{x2 - x1} f(Q_{22}) \tag{3}$$

$$f(x, y) \approx \frac{y2 - y}{y2 - y1} f(x, y1) + \frac{y - x1}{y2 - y1} f(x, y2) \tag{4}$$

Calculating the exact positions of the detected object with (2) to (4) allows a mask to be projected on top of the object. This could particularly be of greater interest in privacy-sensitive issues. The Mask R-CNN algorithm has reached an mAP score of 62.3 percent for an IoU of 0.5 over the 2016 COCO testdev dataset at 5 fps (Ouaknine, 2018; He, et al., 2018). Examples are given in the figure below (Fig. 7.15).

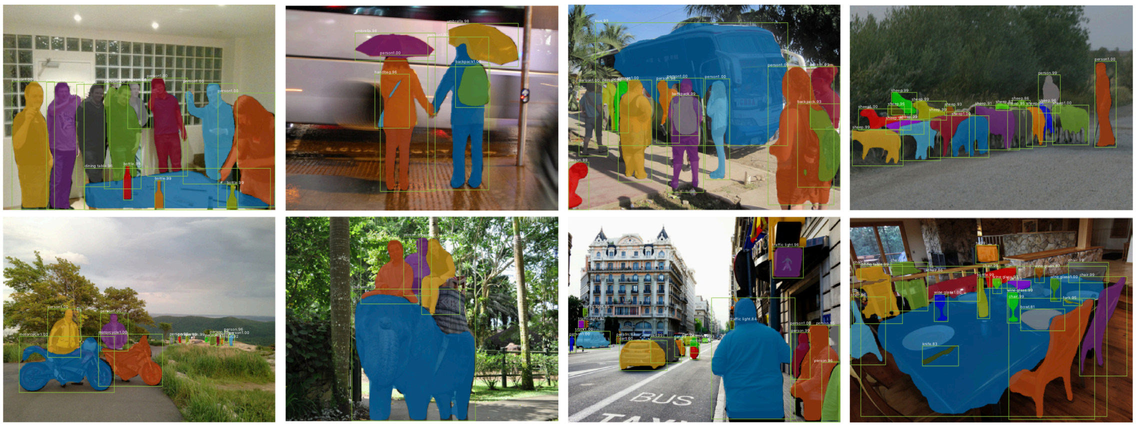

Fig. 7.15: Examples of Mask R-CNN applied to several frames of the COCO test set running at 5 fps (He, et al., 2018)

7.4 Transfer from Medicine

The natural counterpart of artificial neural networks itself comes from natural science and was implemented in computer science as a whole new subject. The transfer was based on basic research in neuroscience and biology (see subchapters 2.1 and 3.2.1). Now, almost three quarters of a century later, biology gets powerful support from deep learning and deep neural network to solve tasks in bioinformatics that are simply too complex, time-consuming, or impossible for a human. One such use case that adapts the Mask R-CNN object detection algorithm to detect exons, is being developed in the Institute of Medical Systems Biology (see subchapter 7.4.1).

7.4.1 Detexon

The protein-coding regions of genes are called exons. Separation of exons can take place by intervening sections of DNA that do not code for proteins, known as introns. Through a process called splicing, the introns are cut out and only the exons remain. Due to the non-biological setting of this thesis, below figure is given to make this subchapter more comprehensible to non-biologists (Fig. 7.16).

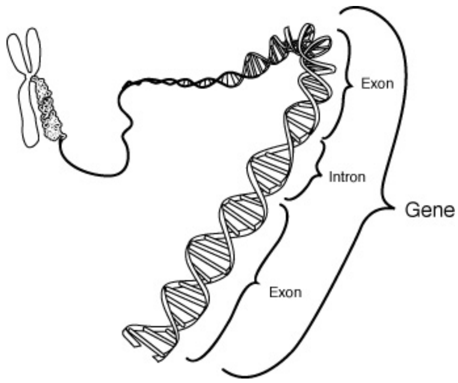

Fig. 7.16: Representation of a chromosome consisting many genes, which again consist introns and exons (Geer, 2001)

Exons consist of the base pairs (Adenine and Thymine, and Cytosine and Guanine) which carry the genetic code of protein synthesis. Mutations result in altering the protein synthesis product which are known to be the cause of many diseases, such as sickle cell anemia, being the result of a single-nucleotide polymorphism mutation. Therefore, given this direct link of a genomic mutation within an exon to disease manifestation, it is of fundamental interest to identify exons within a genome. In this regard, clinical genetic diagnosis heavily relies on exon annotations to develop drugs and therapies (Magruder, 2018b).

On basis of the functional link between exonal mutations and pathology and the lack of exon site prediction and annotation, a deep neural network architecture is proposed by Magruder to predict and discover exons, as well as evaluate the disease relevance. A deep learning approach is backed by the unprecedented amount of unanalyzed data. Therefore, the exon detection network is referred to as Detexon (ibid.).

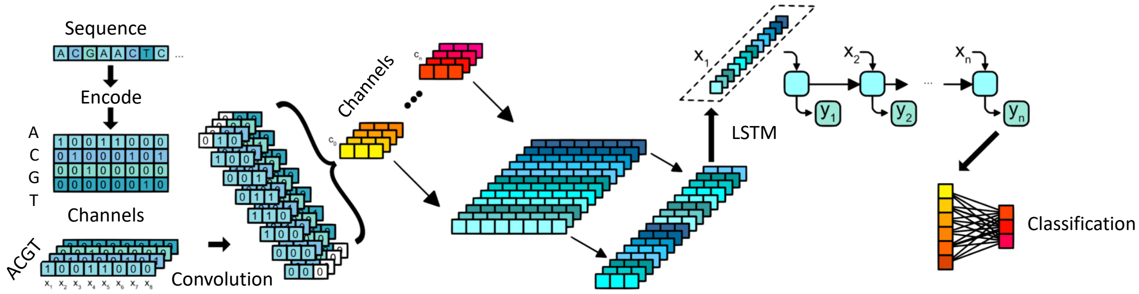

Fig. 7.17: Schematic representation of the used hybrid neural network for exon classification (Magruder, 2018b)

The above figure represents the hybrid neural network, consisting of a CNN that gets the nucleoside chain as input. The input can either be a four-channel time-series or a one-channel four-pixel tall image, each pixel line exclusively representing a nucleoside (Fig. 7.18). The fully connected layer then connects to an LSTM network, where the LSTM cell *reads* the sequence, similar to how polymerase does. In the end, a classification can be made whether the input sequence includes exons or not.

Fig. 7.18: Four-pixel tall image representation of a nucleoside chain; Red arrows mark beginning and end of the detected exon; Bottom figure: cut exon representation, based on (Magruder, 2018c)

The above described classifier, performing with 94 percent accuracy, is currently in development to get included in the YOLO architecture for detection. Additionally, to better predict the exact location of the exon (current solutions perform with a ±10 pixel accuracy), bilinear interpolation used in Mask R-CNN is planned to be included. It is expected that the detection algorithm can detect novel exons in the human genome (Magruder, 2018c).

7.4.2 Transfer to Damage Detection

The technology transfer methodology to find transferable technology described in Chapter 6, in particular subchapter 6.4, is different in this case since the two industries were known beforehand. The main task of this thesis was focused on the finding of transfer possibilities and links between these two industries. However, the application of the methods throughout the work of the thesis clarified the problems and similarities in both fields.

Since the main task of the described use cases in this chapter is to detect objects, be it the precise four-pixel representation of an exon, or a lightning strike damage, it is imaginable, that the current development of Detexon can be adapted to detect damages on aircraft. Although the architecture needs to be that of a standard CNN for two-dimensional images as input, and the need for an LSTM would not be existent, the experience in applying an out of the box solution (YOLO) to a specific problem is valuable. Thus, considering an early technology development stage, Detexon would be rated as on the verge of TRL 4 at this moment, with the integration within the YOLO architecture being in development (see Table A.1). Regarding the classification of the technology readiness, other definitions apply to software according to NASA. The description of TRL 2 to 5 for software is given in the appendix. This evaluation is only possible because of the close work with the Institute of Medical Systems Biology in the course of this thesis.

A *market* and *customer* research on basis of genchi genbutsu was done twice with different focus at the Frankfurt maintenance base of Lufthansa Technik to better grasp the problem of detecting damages. As described in the beginning of this chapter, the investigation was done on-site and with the staff concerned. Maintenance technicians would profit from an augmenting technology, where the high-concentration and exhaustive work in difficult circumstances is done by an intelligent machine. The Delphi method did not come into play in a formal way, since the scope of this thesis does not include the implementation itself. However, experts' opinions always played a role in the investigation as well as evaluation of both use cases, Detexon and AutoInspect. For a more formal approach in accordance with the Delphi method, qualifying experts of both groups would be interviewed extensively.

A patent analysis shows that there are six-figure numbers of patents in gene and genome detection, while only 161 in exon detection (see Table 7.1 below). Of those 161, there are none that use deep learning, but 18 that use a machine learning approach. Generally speaking, deep learning and computer vision are applied to both gene and genome related fields, as well as in medicine and biomedicine. On the other hand, deep learning and computer vision is more used in damage and inspection related topics. Narrowing this down to the aviation and maintenance industry, there are around a thousand patents filed that further deal with detection, of which only 98 are in combination with an unmanned aerial vehicle. From this, it can be derived, that artificial intelligence and deep learning found entry into both the medical and aviation industry, but yet struggles to get implemented in highly specific use cases such as exon detection or drone-assisted visual inspection. It has to be understood, that the implementation of AI and deep learning in particular use cases is important to achieve a higher goal. In aviation, this would be a digital maintenance shop in an industry 4.0 environment, whereas in medicine, the context would be highly-personalized drug development. Both industries therefore can make use of powerful technologies such as deep learning to expand business fields. However, it has to be said that patent analysis is most often done when there is no evident information existent about the industries that use this technology. The thesis started with an assumption that there are similarities between medicine and aviation when it comes to artificial intelligence and deep learning.

Table 7.1: Findings in the patent database of WIPO regarding gene and damage detection keywords

Keyword	**# of patents filed**
(Gene ∨ Genome) ∧ Detection	201,415
Exon Detection	161
(Gene ∨ Genome) ∧ (Deep Learning ∨ Computer Vision)	1,430
(Gene ∨ Genome ∨ Medicine ∨ Biomedicine) ∧ (Deep Learning ∨ Computer Vision)	3,518
Exon Detection ∧ Deep Learning	**0**
Exon Detection ∧ Machine Learning	**18**
(Damage ∨ Inspection) ∧ (Deep Learning ∨ Computer Vision)	6,857
(Damage ∨ Inspection) ∧ (Deep Learning ∨ Computer Vision) ∧ (Aviation ∨ Maintenance)	1,242
(Damage ∨ Inspection) ∧ Detection ∧ (Deep Learning ∨ Computer Vision) ∧ (Aviation ∨ Maintenance)	1,046
(Damage ∨ Inspection) ∧ Detection ∧ (Deep Learning ∨ Computer Vision) ∧ (Aviation ∨ Maintenance) ∧ UAV	**98**

Creativity techniques were applied throughout the whole process of this thesis, especially when extracting use cases in maintenance and the possibilities of application and technology transfer thereof. Divergence happened in various brainstorming sessions at Lufthansa Technik with a preceding introduction to deep learning and deep learning applications and an open discussion about problems in maintenance and aviation in general. Convergence happened right afterwards, with a better understanding of the possibilities of deep learning in the maintenance and aviation environment developed. Moreover, every finding and interesting thoughts were discussed with the medical experts at the Institute of Medical Systems Biology, gaining insights of the problems in a complex real-world environment such as a maintenance shop to adjust the possible realization of the use case. The use case of AutoInspect was chosen because of the similarity in medical detection of exons with only a few adjustments needed.

As already described above, the Detexon algorithm reaches TRL 4, when the implementation inside the YOLO architecture manages to work. This being said, development of AutoInspect could start with the TRL 4 experience of Detexon (see Table A.1). A six months evaluation phase like this thesis could be used as an equivalent to the Research Days described in the chapter before (see subchapter 6.1) to find similarities and transfer links of both industries, as well as to assess the technology's advantages and limits applied in the specific use cases. When coming from an aviation background, problems that occur in that industry are known better, which applies for a target-oriented problem solution. At the end of the six months evaluation phase, a state similar to TRL 2 is expected to be reached.

This proposed method is backed by experience in engineering and technical management as shown below (Fig. 7.19).

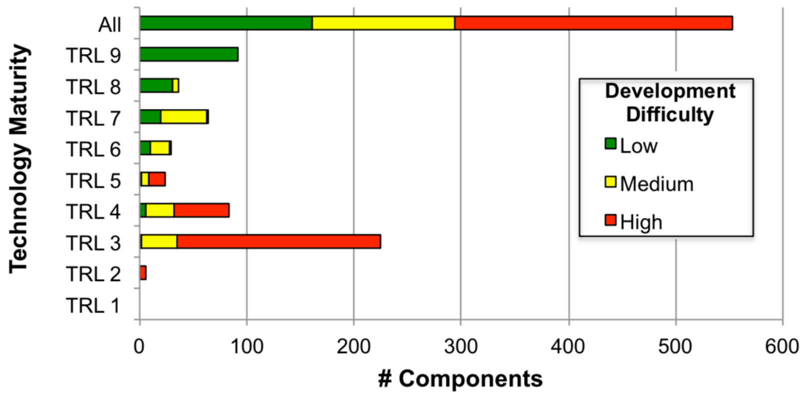

Fig. 7.19: TRL development difficulty representation of a complex project with more than 550 components at BP (formerly British Petroleum); The high number of components result in a more representing average (Olechowski, et al., 2015)

The earlier the development stage of a technology, the more difficult it tends to be, especially for high-number component technologies or projects. The most difficult part is to adopt concepts from the theoretical observation into a real-world use case that defines the value (Olechowski, et al., 2015). The proof of concept for TRL 4, namely a *subsystem validation in a laboratory environment*, is assured by experts in the deep learning field to be adaptable for damage detection with changes to the framework (Bonn & Magruder, 2018). Although the development would be quicker when starting with the basic neural network architecture, the gained knowledge has a certain learning curve, e.g. the development difficulty for AutoInspect drops significantly. The profit in terms of development effort are just assumptive at this point, but definitely existent. Also, the preprocessing and possible data augmentation work prior to the network adjustment still exists, as described in subchapter 7.3.1. Moreover, AutoInspect would most probably work best with an autonomous drone, which lead to further complexity of the whole system. The system development and readiness regarding the autonomous drone development has to be considered beforehand.

7.4.3 Possible Savings and Advantages of a Cooperation

Referring to Fig. 6.1, the question in the red box (*Is discovery sufficiently developed to attract commercial funding?*) could be answered with a *Yes* since the module was extensively developed and successfully tested in a laboratory environment. This would attract possible financial sponsors more easily. A similar use case (here: Detexon) can be shown to financial sponsors or venture capitalists before investing in the technology development in the other industry or environment, respectively. Building on the knowledge of Detexon can most certainly define the innovation as commercially viable, thus saving difficult and time-consuming management decisions which are mostly based on expert's opinions outside the company when developing novel technology, and therefore attribute to cost saving.

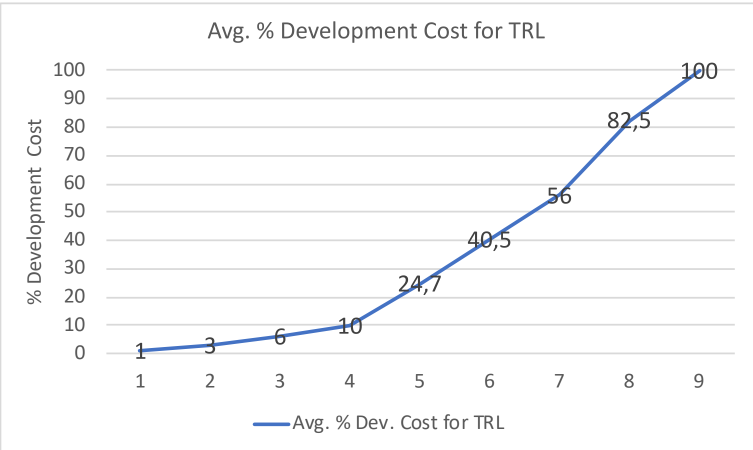

Fig. 7.20: Average percentage of development costs versus TRL (Linick, 2017)

Although the Technology Readiness Level cannot be used directly to estimate costs, there is a relationship between TRL and the percentage of development cost (Linick, 2017). The above figure shows, that the cost development rises exponentially. An adaption from one TRL 4 development to another development and the use of the learning curve (ibid.) would save a portion of the total development cost and effort, looking back to Fig. 7.19 where the largest part of high development difficulty lies in TRL 3 and 4. At this point (see paragraph below), a precise estimation is not possible. However, as already stated, a technology evaluation has to be made beforehand which would add up to the sum, depending on how much rework is done. In addition to that, the labor hours would still apply, although with a slightly faster pace, depending on the experience of the scientist or engineer.

In the course of this thesis, methods of knowledge transfer cost were investigated. More precisely, attempts to estimate development costs with different methods, including Analogy Cost Estimating, Parametric Cost Estimation and Expert Opinion (GAO, 2009, p. 108 ff.; Linick, 2017) were made. In hardware programs, calculations on basis of analogy are a good estimate, because usually, the technical and program definition of the investigated technology program is good enough to make the necessary adjustments. In software however, other parameters, like software size and complexity, are relevant, and assumptions need to be proven stricter, e.g. the reuse of code is a common pitfall for engineers. Overall the methods for software are still immature, therefore, a more detailed cost analysis is needed (GAO, 2009, p. 125 ff.). However, this would be out of the scope of this thesis.

One of the more easier methods regarding analog cost estimation would be possible by having one representative system parameter in the development program. The ratio of both parameters would be used to form a factor, with which the development costs of a certain TRL could be estimated from the program costs of the known program (Linick, 2017; see Fig. 7.20). But especially for artificial neural networks, a scientific guideline to set hyperparameters for the estimation of network complexity, network size, or the development effort are simply nonexistent, yet a famous research problem (Magruder, 2018d; Bergstra, et al., 2013). A neural network contains a large number of parameters as already described in previous chapters. Thus, from the previous work in the field of Detexon, theoretically *only* a learning curve can be applied for further adaptions in the case of AutoInspect. An accelerated development time for a second task or project of ten percent is usually the case (Mályusz & Varga, 2017). Though, expert opinions on this matter have shown that the majority of the time is spent preprocessing and bringing the data into a usable shape. The adaption part of the basic neural network architecture like YOLO or Mask R-CNN is not challenging in relation to the preliminary work (Magruder, 2018d).

Cost drivers would be the hardware (GPUs, Computer, Cloud Computing), the electricity for it, the IT infrastructure, coding software, and staff. A deep learning expert would need less than a month for the implementation of such a network, after evaluating the specific industrial needs, with possibly a finetuning afterwards. But that is the crux: Deep learning experts are a scarce resource. According to recent reports, many deep learning scientists with a PhD in deep learning get salaries from medium six to low seven figures a year due to industry competition (Metz, 2017; 2018; Yakovenko, 2018). Notably most of the companies that are in the AI race are located in the United States and China, with Europe on the verge of setting-up (EC, 2018). Opportunities to bring deep learning knowledge inside the company is either by collaborating with science labs and institutes that educate fresh PhD's, or to pay large sums to a deep learning expert, that are highly wanted at the moment. Therefore, deep learning prospectively remains a wanted domain for the next few years, where a PhD education might be a good option for low to intermediate priority development projects. A development project like AutoInspect would take a PhD student about two years to fully implement (Magruder, 2018d). The sole cost, when cooperating with a science lab or institute, would be around €100.000 for the course of two years.

7.5 Further Concept Features

To solve the problem of autonomous visual inspection, there is the need to use further technologies, namely autonomous unmanned aerial vehicles, that can assist in achieving the objective. There are other ways of collecting damage data, though only robotics make sense when considering process acceleration and automation. A quadcopter, or any other type of flying drone, would offer the needed flexibility to implement an autonomous system inside a maintenance hangar as well as outside on the apron of the airport. However, the development time and system complexity also rise with the number of interacting subsystems. Autonomous unmanned aerial vehicles just like autonomous land vehicles usually include sensors to detect its environment, including radar sensors, camera systems and LIDAR (light detection and ranging), of which some might also be implemented redundantly (Santo, 2016).

Another point worth mentioning is the online detection function, e.g. a steady and high enough frame rate to guarantee real-time tracking. Although having a high frame rate is not as critical in damage detection on aircrafts as it is in autonomous vehicles, a smooth frame rate would allow maintenance technicians or engineers to track the detection process in real-time or revisit places of uncertainty. This might also be helpful in cases when there is the need for an experienced maintenance or structure engineer to assist the detection process and initiate further steps. As described in subchapter 7.1, the current communication channel comprises e-mail and telephone. However, having an online function with a high frame rate, e.g. 30 to 60 fps, with a high enough accuracy at the same time, requires high computational power (Ragate & Asai, 2017). GPUs have shown to perform much better than multi-core CPUs (see subchapter 7.3.3), although there are developments for non-GPU computation. In particular, a novel architecture development called Fast YOLO performs with 3.3 times higher frame rates on embedded systems compared to the original YOLO v2 (Shafiee, et al., 2017). In case the accuracy allows it, drones could inspect aircrafts off-site without the need for extra computational power. However, this field is still relatively new and in need of further research.

To better visualize the findings and allow an easier way of communication, a three-dimensional mapping of the damages can be realized with the collected data. Since with detection there comes a location information, usually the position of the upper left corner of the box, it can be matched with the local positioning of the drone. In addition of GPS-enabled drones, indoor localization is also possible with the right settings, e.g. radio-based localization systems (Stegagno, 2018).

8 Conclusion

In the course of a few years, deep learning has shown to be a highly potential technology, solving complex tasks that were simply not solvable with this level of accuracy. To prove this, parts of the introduction to artificial intelligence (Chapter 2) have been analyzed with natural language processing (Siri) in German. Apple's Siri processed and transcribed 8947 words with an error rate of 4.325 percent. The transcribed text then was again translated with a deep learning machine translation tool (www.deepl.com) from German to English. The machine translated part counted 3229 words with an error rate of 1.17 percent, which is significantly below a human's error rate of about 5 percent. Deep learning thus has shown to perform better than human-level in some specific tasks. In image recognition and detection, deep learning algorithms have shown to perform with similar error rates (see subchapter 4.1.2). To better visualize the findings, a SWOT analysis regarding the technology of deep learning is represented in below figure (Fig. 8.1).

Strengths - Time and Cost effective, low error - Prediction, Detection, Natural Language, Genetic Algorithms, and more, all within same language - Ability to analyze big data - Customizable for various tasks - Enabling new business models	Weaknesses - No expertise (experts are highly expensive right now and have freedom to choose) - Understanding the inner computation difficult (as of now) - Data hungry
Opportunities - Identification of use cases and manual processes (direct savings) - Cooperation with an expert group or research institute - European and state fundings	Threats - Aviation regulations regarding artificial intelligence - Error rate not small enough for aviation purposes

Fig. 8.1: SWOT analysis of deep learning technology in aviation

Regarding tasks that require deep learning to be solved, the bottleneck lies outside the deep learning framework and algorithm; most often the data is the constraint. Before that, the biggest constraint was computational power. In the case of object recognition using CNN's, data acquisition, preprocessing, cropping and getting the data in the right and usable format to train the network take most of the time. In the case of damage detection on aircrafts this would mean a standardized process for image taking over a set period to gather a solid amount of data for training the network. Training the network with the gathered data from Lufthansa Technik's maintenance site in Frankfurt is possible but linked to a huge effort in data preprocessing.

To make the implementation easier and optimize the cost structure, analogies in medical use cases were sought, where object detection was used to locate exons in genomes. Although the transfer methodology of novel technology development < TRL 4 was researched, the common deep learning frameworks are generalized to a certain degree and adaptable for every kind of task, irrelevant of the investigated industry. However, there were found links between aviation and medicine. Similarities include human factors, e.g. sentiment analysis and behavior analysis to detect tiredness of pilot and technicians in a highly exhaustive work environment, human support, e.g. analyzing data much faster in irregular working conditions like damage detection on the apron, and other use cases mentioned in Chapter 5 to 7.

However, the technology transfer process described in this thesis can help to build bridges between different industries that want to make use of artificial intelligence and deep learning. In fact, the thesis itself can be seen as a transfer instrument between industries similar to what is practiced in state-of-the-art technology transfer processes (see subchapter 6.4). It is therefore notable, that different industries like medicine and aviation can have commonalities in the use of the same technology with similar use cases. Matching the industry and problem specific expertise of one party with the solution application skills of another, cross-industry projects can have a beneficial outcome in early technology development, like seen in Chapter 7.

Of the technology transfer from exon detection to damage detection however, only the learning curve of the medical framework could be of use. After all, basic frameworks of object detection are in constant development and ready to use with an environmental investigation (analog to TRL 2). The case of Detexon was a special one and might not be the best to choose for a direct transfer to an aviation problem, since it uses an additional LSTM network. It is possible to downgrade the whole network to the parts needed for damage detection but realistically, it would be time-saving to start on the basic framework of YOLO or Mask R-CNN. What would be possible is to take classifier frameworks that are also developed in the Institute of Medical Systems Biology that classify on basis of iris scans how the probabilities for certain diseases are (similar to subchapter 5.1.1), and combine it with a detection framework. It is expected that there is an estimation of seven percent savings for the use of the similar development of the deep learning system (from TRL 2 to TRL 4). However, the development of the deep learning system in the early stage is in terms of effort not as much as the later stages, where it has to pass the field test. Applying the frameworks, which are documented open-source, thus applicable for a scientist with deep learning expertise, is not difficult. Yet, with the hype still going on about deep learning, it is difficult to find experts in deep learning without investing a relatively high sum to stay in the competition of applying cutting-edge AI technology to business. This being said, it is not of importance of which background the deep learning expert comes from.

Although methodologies of (analog) cost estimation for early stage technology development were reviewed, they could not be applied due to the unknown relevant parameters in a deep neural network. Moreover, the parameters are constantly changed since there is no known methodological and scientific approach to set the hyperparameters of a deep neural network. Every neuron has several underlying parameters, which add up to the sum of millions of parameters, in addition to the hyperparameters of the whole network. This is also one of the reasons deep learning is not implemented in many safety-related domains since the way neurons work in a complex network cannot be specified besides a high-level understanding of the feature extraction in the case of object recognition or detection for example. The following quote therefore reflects the complexity of artificial neural networks:

> *Will we understand how such intelligent (deep neural) networks work? Perhaps the networks will be opaque to us, with weights and biases we don't understand, because they've been learned automatically. In the early days of AI research people hoped that the effort to build an AI would also help us understand the principles behind intelligence and, maybe, the functioning of the human brain. But perhaps the outcome will be that we end up understanding neither the brain nor how artificial intelligence works!*
>
> *– Nielsen, 2017*

8.1 Recommendation for Action

Since the topic of deep learning is still relatively new, research is inevitable for a company that wants to apply state of the art technology. This subchapter covers reasons based on the written thesis why investments in deep learning should be considered right now.

Deep learning is a highly transferrable technology, therefore a recommendation on basis of the findings of this thesis would be to begin with a pilot project as soon as possible. If in the course of the project the use case turns out to not be financially viable, the use case can be altered with little adjustments to the model if the problem is similar. A broad framework for the pilot project should be provided. Therefore, smaller investments for work periods of six to nine months are not sufficient to get satisfactory results. In most cases, the issue of whether the concept performs as promised dictates the taken actions. On basis of the results of this thesis, deep learning proof of concepts can be generated within a short period by junior researchers or graduates. Undergraduates in computer science, engineering or computer engineering might be solution in cooperation with a research institute, especially since the investment risk is low. The requirements would be to be familiar with the theoretical foundations of deep learning, corroborate coding skills, and most importantly show proficiency in problem solving. Knowing the relevant environment would be an extra advantage. This could be combined with technology transfer instruments that are being used to transfer knowledge from academia to the private sector to further attract interest inside the company for the new technology.

Increasing rivalry in the industry makes research in deep learning inevitable to keep a competitive advantage. The focus should lie on developing new opportunities as well as support current business. According to the three growth horizons, a company should strategically place its business fields in every horizon to ensure competitiveness. Research and early technology development without doubt is placed in the last horizon (see Fig. 8.2), creating options for future business (Baghai, et al., 1999).

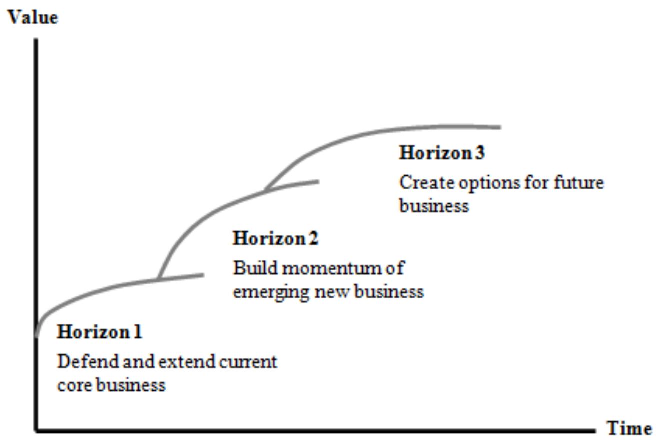

Fig. 8.2: McKinsey’s three growth horizons (Baghai, et al., 1999)

However, it is expected that business fields in other horizons can highly profit from artificial intelligence and deep learning. According to market studies, not only tech giants like Apple (NLP, voice recognition, face recognition) or Google (several, including deep reinforcement learning), or startups like Uber (self-driving vehicles, trip forecasting) and Netflix (recommending system) profit from intelligent products and services, but also more conservative companies like GE apply deep learning to their products (medical imaging, in cooperation with NVIDIA) (Sergeev & Del Balso, 2017; Seville Report, 2017; Oppermann, 2018). What these mentioned companies have in common are their high investments in R&D. Spending $1.01 billion in the last quarter of 2012, Apple’s R&D expenses were constantly growing to $3.407 billion quarterly by end of 2017 (YCharts, 2018). Similarly, this is the case with all the above-mentioned companies.

On basis of interviews with the central innovation department, and work experience in the product division of Original Equipment Innovation, a clear-cut distinction between machine learning and deep learning has to be made. Since these two words, along with other hyped technologies like blockchain and IoT, are often used buzz words in managerial circles, a precise definition and understanding regarding new technology has to be met. Technology reviews should be made available regularly for employees and managers in an easy to digest format. This would include cooperations with technological institutes as described in subchapter 6.1. Participating and holding innovation days one time a year is not sufficient; other transfer instruments should be intensively reviewed.

Especially given the fact that the industry 4.0 environment is around the corner and many companies prepare for it, Lufthansa Technik has to take every opportunity possible to direct its business fields towards the digital age.

8.2 Outlook

Deep learning is still a highly researched topic. In terms of up-to-dateness, deep learning is on its highest point of public awareness right now (Fig. 8.3). As seen in Fig. 1.1, deep learning was near the highest point of the 2017 hype cycle. Of course, time will say if the predicted dimensions are true, both the optimistic growth as well as the inflated expectations, but in light of this thesis, deep learning can enable many cross-industrial use cases as of now.

Fig. 8.3: Qualitative Google Trends search results regarding *Deep Learning* (Google, 2018)

A further point that could not be investigated due to insufficient research on this topic, is how a deep neural network model can be weighed in terms of complexity. As of now, there was no representing parameter found that could be used as a comparison for neural networks. This step is just the beginning of trying to understand deep neural networks since one of the reasons it is not being actively considered in aviation, besides not having the needed expertise, is the inability to understand how the model internally works (Stövesand, 2017). This was also the reason why no analog cost estimation could be done for the adaption of the used techniques in Detexon for AutoInspect. The estimation of ten percent profit due to the learning curve is not verifiable at this stage and should be proven in a further research. Building on top of TRL 4 when a use case with similar network topology is found, can save up to seven percent in development cost of the deep learning system.

Another point worth mentioning, and interesting as a further research topic, is the system complexity and development of an autonomous unmanned aerial vehicle in a maintenance environment as well as the human-machine interaction. The interplay of subsystems is expected to be the most difficult part since eventualities are not foreseeable. The technology development of the whole system would also be more accurate to do cost estimations than simply the deep learning subsystem. The integration inside the company's existing procedural landscape is expected to not be as easy as implementing it in a new environment. After all, both inspection forms will run simultaneously until the new system has proven its long-term performance ability since maintenance of aircrafts is one of Lufthansa Technik's main business fields. However, its contribution to the company's savings is not negligible. The exact numbers would need to be calculated or estimated in further research work.

This thesis however tried to outline the basic definitions and possibilities of machine learning and deep learning as well as the differences between these two subsets of artificial intelligence. It was aimed to present an introduction to these matters to boost the analysis of further use cases regarding deep learning. It is hoped that this thesis be of use in further graduate and research projects and to be of benefit to Lufthansa Technik to advance digital business in the aviation industry and contribute to the implementation of artificial intelligence, and work towards the next step – artificial general intelligence.

Appendix

In this last part, all other figures, tables and source codes that were redundant or additional and would not contribute to a better understanding of the matter, are presented here.

Figures

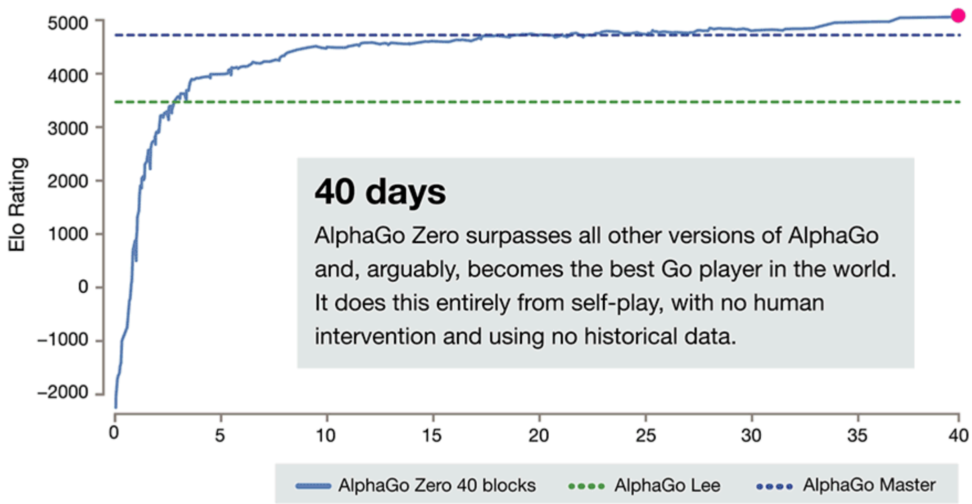

Fig. A.1: Learning curve of AlphaGo Zero in self-play (Hassabis & Silver, 2017)

Tables

Table A.1: Explanation of TRL 2 – 5 in software according to DoD / NASA (Majumdar, 2007)

TRL	Definition	Description	Support Information
2	Technology concept and/or application formulated	Once basic principles are observed, practical applications can be invented. Applications are speculative, and there may be no proof or detailed analysis to support the assumptions. Examples are limited to analytic studies using synthetic data	Applied research activities, analytic studies, small code units, and papers comparing competing technologies
3	Analytical and experimental critical function and/or	Active R&D is initiated. The level at which scientific feasibility is demonstrated through analytical and	Algorithms run on a surrogate processor in a laboratory environment, instrumented

	characteristic proof of concept	laboratory studies. This level extends to the development of limited functionality environments to validate critical properties and analytical predictions using nonintegrated software components and partially representative data	components operating in laboratory environment, laboratory results showing validation of critical properties
4	Module and/or subsystem validation in a laboratory environment (i.e., software prototype development environment)	Basic software components are integrated to establish that they will work together. They are relatively primitive with regard to efficiency and robustness compared with the eventual system. Architecture development initiated to include interoperability, reliability, maintainability, extensibility, scalability, and security issues. Emulation with current/legacy elements as appropriate. Prototypes developed to demonstrate different aspects of eventual system	Advanced technology development, standalone prototype solving a synthetic full-scale problem, or standalone prototype processing fully representative data sets
5	Module and/or subsystem validation in a relevant environment	Level at which software technology is ready to start integration with existing systems. The prototype implementations conform to target environment/interfaces. Experiments with realistic problems. Simulated interfaces to existing systems. System software architecture established. Algorithms run on a processor(s) with characteristics expected in the operational environment	System architecture diagram around technology element with critical performance requirements defined, Processor selection analysis, Simulation / Stimulation (Sim/Stim) Laboratory buildup plan. Software placed under configuration management, COTS/GOTS in the system software architecture are identified

Source Codes

Digital Image Processing (Matlab)

```
Prep=imread('imagergb.jpg'); % inputs damage image

intval=uint8(zeros(size(Prep,1),size(Prep,2)));
% gets horizontal and vertical size of the image

for i=1:size(Prep,1)
      for j=1:size(Prep,2)
intval(i,j)=0.1989*Prep(i,j,1)+0.4870*Prep(i,j,2)+0.0140*Prep(i,j,3);
      end
end
% pixel-wise for-loop to adjust intensity of each channel

out1 = imadjust(intval,[0.46 0.52],[])
out2 = imadjust(intval,[0.4 0.45],[])
% adjustment of intensity values to a new range

figure;
subplot(1,3,1)
imshow(Prep)
title('No changes');
subplot(1,3,2)
imshow(out1)
title('Medium feature extraction');
subplot(1,3,3)
imshow(out2)
title('Maximum feature extraction'); % output of all three figures
```

Assigning Colors to Pixel Values (Excel Macro)

```
Public Sub AssignColor()
  Dim cel As Range
  Dim myRange As Range

  Set myRange = Range("A1:AY25")
  For Each cel In myRange
    cel.Interior.Color = RGB(cel.Value, cel.Value, cel.Value)
    If cel.Value > 127 Then
      cel.Font.Color = RGB(0, 0, 0)
    Else
      cel.Font.Color = RGB(255, 255, 255)
    End If
  Next cel
End Sub
```

Research Proposal

Following, Chapter 8 (Structure) and 9 (Schedule) of the research proposal made on basis of this thesis for further investigation of this problem is presented. Since the research proposal is not part of the thesis, and were originally written in German, an online deep learning machine translation tool (www.deepl.com) is used to translate the text into English, without further proofreading or correction. Doing so, the advances of deep learning in real-world applications are again shown.

Structure

The structure of the doctorate and the timetable (Chapter 9) are not identical in chronological order. While the structure rather represents the monographic division of the thematic chapters, further (preliminary) works can be found which are not listed here. In order to be able to formulate a universality of the research results, attention is paid from the outset to the learning of generalized concepts. In order to increase the transfer of learning and to increase the yield of knowledge, research is also conducted on advanced methods and concepts that are not listed in this structure.

1. Introduction: description of the subject(s), classification of the work, explanation of the problem, background and motivation
2. State of the art: Previous work in the fields of autonomous vehicles, in particular autonomous aircraft, deep learning, in particular object detection and associated algorithms, human-machine interfaces and interaction, proof of concept of damage detection / visual inspection (master thesis)
3. Objective of the promotion: Development of new technologies, automation in maintenance, process acceleration, on-site maintenance, digitization of maintenance results, new maintenance concepts, integration into an industry 4.0 process landscape (bigger picture)
4. Theoretical study: modelling of the Autonomous Aircraft, modelling of damage detection, data preparation, training of models, simulation
5. Experimental investigation: (statistical) design of experiments, testing on the real example and execution of the experiment, documentation and evaluation, comparison/comparison with theory and simulation, interpretation of the results
6. Summary and findings: Exploitation of the results, further tasks and open points, outlook, further research questions

The engineering nature of the doctoral project suggests a monograph rather than cumulative publications in the course of the project.

Translated with www.DeepL.com/Translator

Schedule

The following schedule presents a rough list of the activities within the scope of the doctoral project. Most of the works listed are sorted chronologically, but may vary depending on the results and environmental conditions. A visual schedule is given in below figure (Fig. A.2).

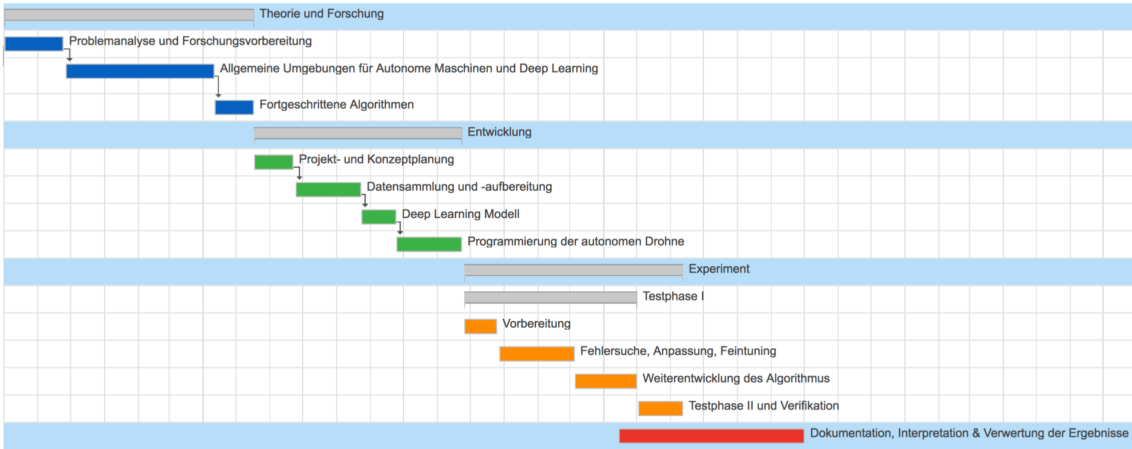

Fig. A.2: Visual schedule of the research proposal; in German

- Problem analysis, estimation of demand and workload - 80h
- Problem and solution interviews - 60h
- Preparation of the programming environment in Python - 80h
- General and Applied Autonomous Drones Environments - 400h
- General Machine Learning & Deep Learning Environments - 300h
- Advanced object recognition algorithms and techniques - 200h
- Project planning and management according to Lean - 80h
- Investigations and analysis of the application site, subsequent concept planning and general conditions - 120h
- Data collection, maintenance and processing, preprocessing, - 320h
- Deep Learning Modelling and Training - 160h
- Programming of the autonomous drone - 320h
- Test phase I and adjustment, incl. lead time and planning of maintenance times - 160h
- Probable error search and correction, adaptation of the model and the algorithm, fine tuning - 200h
- Further development of algorithms for object detection, hybrid models, increase of speed and accuracy, test experiments with data sets - 300h

- Test phase II and adjustment, incl. lead time and planning of maintenance times - 160h
- Documentation, interpretation of results, writing process, publications - 900h
- Optional: Further training in related subject areas, depending on specialisation and focus decentralised applications, decentralised artificial intelligence, machine-to-machine payment, further industry 4.0 and production or machine shop concepts to initiate further research topics.

Translated with www.DeepL.com/Translator

List of References

5G PPP, 2017. *5G PPP.* [Online]
Available at: https://5g-ppp.eu/wp-content/uploads/2017/07/5G-PPP-5G-Architecture-White-Paper-2-Summer-2017_For-Public-Consultation.pdf
[Accessed 24 September 2017].

AAU, n.d. *Aalborg University.* [Online]
Available at: http://www.en.patent.aau.dk/technology-transfer-process/
[Accessed 11 March 2018].

Ackert, S., 2010. *Aircraft Monitor.* [Online]
Available at:
http://aircraftmonitor.com/uploads/1/5/9/9/15993320/basics_of_aircraft_maintenance_programs_for_financiers___v1.pdf
[Accessed 7 April 2018].

Agesscene, 2009. *Wikimedia.* [Online]
Available at: https://commons.wikimedia.org/wiki/File:Tic-tac-toe-full-game-tree-x-rational.jpg
[Accessed 10 January 2017].

Alexander, C., 2017. *ICEAA Online: Parametric Cost and Schedule Modeling for Early Technology Development.* [Online]
Available at: http://www.iceaaonline.com/ready/wp-content/uploads/2017/07/DA07-Paper-Alexander-Parametric-Modeling-for-Early-Tech-Development.pdf
[Accessed 4 May 2018].

Alvarez, W., 2017. *Wikimedia.* [Online]
Available at: https://commons.wikimedia.org/wiki/File:Markov_Decision_Process.svg
[Accessed 22 April 2018].

Anomaly, A., 2016. *Stack Overflow.* [Online]
Available at: https://stackoverflow.com/questions/40537503/deep-neural-networks-precision-for-image-recognition-float-or-double
[Accessed 4 September 2017].

Arlen, T. C., 2018. *Medium: Understanding the mAP Evaluation Metric for Object Detection.* [Online]
Available at: https://medium.com/@timothycarlen/understanding-the-map-evaluation-metric-for-object-detection-a07fe6962cf3
[Accessed 3 May 2018].

ASB, 2007. *FAA Aircraft Certification.* [Online]
Available at: http://www.faa-aircraft-certification.com/43-9-content-maintenance-records.html
[Accessed 15 April 2018].

Asimov, I., 1982. *Meine Freunde, die Roboter.* 1. ed. Dresden: Heyne Verlag.

Autodesk, n.d. *Autodesk.* [Online]
Available at: https://www.autodesk.com/customer-stories/airbus
[Accessed 23 September 2017].

Baghai, M., Coley, S. & White, D., 1999. *Die Alchimie des Wachtums: Die McKinsey-Strategie für nachhaltig profitable Unternehmensentwicklung.* 1. ed. London: Econ Verlag.

Baklitskaya, N., 2016. *Accenture Technology Labs Blog.* [Online]
Available at: https://www.accenture.com/us-en/blogs/blog-generative-design-optimization-creativity-best
[Accessed 23 September 2017].

Barmaksiz, V., 2018. *Interview mit der Technologietransferabteilung des DESY* [Interview] (10 March 2018).

Baroni, M. et al., 2017. *Open Review: COMMAI: EVALUATING THE FIRST STEPS TOWARDS A USEFUL GENERAL AI.* [Online]
Available at: https://openreview.net/pdf?id=Syh_o0pPx
[Accessed 9 May 2018].

Barrett, R., 2017. *Healthdata Management.* [Online]
Available at: https://www.healthdatamanagement.com/news/sutter-health-ibm-making-progress-on-heart-failure-prediction-model
[Accessed 13 March 2018].

Beck, F., n.d. *Neuronale Netze.* [Online]
Available at: http://www.neuronalesnetz.de/verbindungen.html
[Accessed 4 September 2017].

Bellamy, W., 2017. *Aviation Today.* [Online]
Available at: http://www.aviationtoday.com/2017/09/08/tap-portugal-talks-affordable-ifc-europe/
[Accessed 14 September 2017].

Bengio, Y. & LeCun, Y., 2007. *Scaling Learning Algorithms towards AI.* [Online]
Available at: http://yann.lecun.com/exdb/publis/pdf/bengio-lecun-07.pdf
[Accessed 20 April 2018].

Bergstra, J., Yamins, D. & Cox, D. D., 2013. Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. *JMLR: W&CP,* Issue 28.

Bishop, C. M., 2009. *Pattern Recognition and Machine Learning.* 8. ed. New York: Springer Science+Business.

Blackmore, S., 2010. *Conciousness: An Introduction.* 3 ed. Oxon: Routledge.

BMVA, n.d. *British Machine Vision Association and Society for Pattern Recognition.* [Online]
Available at: http://www.bmva.org/visionoverview
[Accessed 20 February 2018].

Bolonkin, A., 2011. *Universe, Human Immortality and Future Human Evaluation.* 1. ed. Oxford: Elsevier.

Bonn, S. & Magruder, S., 2018. *Interview regarding the possibility of implementing an autonomous visual inspection architecture on already developed frameworks* [Interview] 2018.

Borge, L. & Bröring, S., 2017. Exploring effectiveness of technology transfer in interdisciplinary settings: The case of the bioeconomy. *Creative Innovation Management,* Issue 26, pp. 311-322.

Bossmann, J., 2016. *World Economic Forum.* [Online]
Available at: https://www.weforum.org/agenda/2016/10/top-10-ethical-issues-in-artificial-intelligence/
[Accessed 6 February 2018].

Boyd, C., 2018. *The Startup.* [Online]
Available at: https://medium.com/swlh/the-past-present-and-future-of-speech-recognition-technology-cf13c179aaf
[Accessed 21 February 2018].

Britz, D., 2015. *Wild ML: Artificial Intelligence, Deep Learning and NLP.* [Online]
Available at: http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/
[Accessed 18 April 2018].

Brownlee, J., 2016. *Machine Learning Mastery.* [Online]
Available at: https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/
[Accessed 31 January 2018].

Burger, C., 2016. *Tastehit.* [Online]
Available at: https://www.tastehit.com/blog/google-deepmind-alphago-how-it-works/
[Accessed 11 January 2017].

Casey, B., 2013. Securing Your Future with Proactive Maintenance. *Machinery Lubrication*, 2.

Castelvecchi, D., 2017. *Nature.* [Online]
Available at: https://www.nature.com/news/astronomers-explore-uses-for-ai-generated-images-1.21398
[Accessed 22 April 2018].

Chaudhri, V., 2011. *Stanford University.* [Online]
Available at: https://web.stanford.edu/class/cs227/Lectures/lec01.pdf
[Accessed 21 February 2018].

Che-Wei, 2016. *Instructables.* [Online]
Available at: http://www.instructables.com/id/How-to-Design-a-Bike-Stem-in-Dreamcatcher/
[Accessed 18 Juli 2017].

Choi, E., Schuetz, A., Stewart, W. F. & Sun, J., 2017. *Georgia Institute of Technology.* [Online]
Available at: https://arxiv.org/pdf/1602.03686.pdf
[Accessed 13 March 2018].

Chon, A., Balachandar, N. & Lu, P., 2017. *Deep Convolutional Neural Networks for Lung Cancer Detection,* Stanford: Stanford University.

Chui, M., Manyika, J. & Miremadi, M., 2016. *McKinsey & Company.* [Online]
Available at: https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/where-machines-could-replace-humans-and-where-they-cant-yet
[Accessed 6 February 2018].

Coberly, C., 2018. *Techspot.* [Online]
Available at: https://www.techspot.com/news/74503-fatal-autonomous-uber-crash-could-have-result-improper.html
[Accessed 9 May 2018].

Colceriu, C., 2016. *Krones Blog.* [Online]
Available at: https://blog.krones.com/blog/technologie/bionik-die-zukunft-des-maschinenbaus/
[Accessed 23 September 2017].

Cressey, D., 2008. *Nature.* [Online]
Available at: https://www.nature.com/news/2008/080317/full/news.2008.676.html
[Accessed 18 March 2018].

Da Vinci, L., 2005. *Modell eines humanoiden Roboters.* [Art] (Leonardo da Vinci: Mensch - Erfinder - Genie Ausstellung).

Dalí, S., 1931. *Die Beständigkeit der Erinnerung.* [Art] (Gala Salvador Dalí Foundation).

De Kock, 2003. *University of Pretoria.* [Online]
Available at: https://repository.up.ac.za/bitstream/handle/2263/22959/06Chapter6.pdf
[Accessed 21 February 2018].

DE, 2017. *Digital Engineering.* [Online]
Available at: http://www.digitaleng.news/de/baker-hughes-develops-predictive-maintenance-software-gas-oil-extraction-equipment-using-data-analytics-machine-learning/
[Accessed 8 April 2018].

Dernoncourt, F., 2015. *Stackexchange.* [Online]
Available at: https://stats.stackexchange.com/questions/146413/why-convolutional-neural-networks-belong-to-deep-learning
[Accessed 17 April 2018].

Descartes, R., 1870. *René Descartes´ Philosophische Werke.* 1 ed. Berlin: Heimann.

DIAG, 2013. *Diagnijmegen.* [Online]
Available at: http://www.diagnijmegen.nl/index.php/Featured_Image_Archive
[Accessed 24 March 2018].

Disup, 2016. *Dis-up!.* [Online]
Available at: http://www.disup.com/horus-dispositivo-ciegos-italia/
[Accessed 24 March 2018].

Dongens, N., 2018. *Machine Learning Blog.* [Online]
Available at: https://machinelearning-blog.com/2018/02/21/recurrent-neural-networks/
[Accessed 17 April 2018].

Dvorak, P., 2016. *Windpower Engineering & Development.* [Online]
Available at: https://www.windpowerengineering.com/business-news-projects/offshore-wind-innovation-steep-growth-curve/
[Accessed 28 April 2018].

Dyer, C., 2017. *Github.* [Online]
Available at: https://github.com/oxford-cs-deepnlp-

2017/lectures/blob/master/Lecture%202b%20-%20Overview%20of%20the%20Practicals.pdf
[Accessed 13 April 2018].

EAI, n.d. *Experimental Aircraft Info.* [Online]
Available at: https://www.experimentalaircraft.info/articles/aluminum-corrosion.php
[Accessed 8 April 2018].

EC, 2018. *European Commission.* [Online]
Available at: https://ec.europa.eu/digital-single-market/en/artificial-intelligence
[Accessed 6 May 2018].

El Sallab, A., Abdou, M., Perot, E. & Yogamani, S., 2017. *Arxiv: Deep Reinforcement Learning framework for Autonomous Driv- ing.* [Online]
Available at: https://arxiv.org/pdf/1704.02532.pdf
[Accessed 22 April 2018].

Elgammal, A., Liu, B., Elhoseiny, M. & Mazzone, M., 2017. *Arxiv: CAN: Creative Adversarial Networks Generating "Art" by Learning About Styles and Deviating from Style Norms.* [Online]
Available at: https://arxiv.org/pdf/1706.07068.pdf
[Accessed 9 May 2018].

Esteva, A. et al., 2017. Dermatologist-level classification of skin cancer with deep neural networks. *Nature,* 542(7639), pp. 115-118.

Excell, J., 2016. *The Engineer.* [Online]
Available at: https://www.theengineer.co.uk/device-for-the-blind-uses-computer-vision-and-machine-learning/
[Accessed 24 March 2018].

Fernando, C. et al., 2017. *Arxiv: PathNet: Evolution Channels Gradient Descent in Super Neural Networks.* [Online]
Available at: https://arxiv.org/pdf/1701.08734.pdf
[Accessed 9 May 2018].

Fine, J., 1997. *Georgetown University.* [Online]
Available at: http://faculty.georgetown.edu/bassr/511/projects/letham/final/chess.htm
[Accessed 21 August 2017].

Finn, C., 2017. *Berkeley Artificial Intelligence Research.* [Online]
Available at: http://bair.berkeley.edu/blog/2017/07/18/learning-to-learn/
[Accessed 23 April 2018].

Forsberg, D., 2015. *Avolon.* [Online]
Available at: http://avolon.aero/wp/wp-content/uploads/2015/03/Avolon-White-Paper-

FInal-30-March-2015.pdf
[Accessed 6 April 2018].

Fortuner, B., 2018. *Github of Brendan Fortuner.* [Online]
Available at: https://github.com/bfortuner/ml-cheatsheet/blob/master/docs/architectures.rst
[Accessed 21 April 2018].

Freitag, L., 2016. *Wirtschafts Woche.* [Online]
Available at: http://www.wiwo.de/unternehmen/industrie/industrie40/big-data-wer-die-zahlen-hat-hat-die-macht-/12918296.html
[Accessed 26 August 2017].

Fromm, E., 2015. *Sigmund Freuds Psychoanalyse – Größe und Grenzen: Greatness and Limitations of Freud's Thought.* 1. ed. s.l.:Open Publishing Rights.

GAO, 2009. *GAO Cost Estimating And Assessment Guide Best Practices for developing and managing Capital Program Costs,* United States: United States Government Accountability Office.

Gardiner, G., 2016. *Composites World.* [Online]
Available at: http://www.compositesworld.com/blog/post/bionic-design-the-future-of-lightweight-structures
[Accessed 15 07 2017].

Garg, N., 2015. *Quora: What is the vanishing gradient problem?.* [Online]
Available at: https://www.quora.com/What-is-the-vanishing-gradient-problem?utm_medium=organic&utm_source=google_rich_qa&utm_campaign=google_rich_qa
[Accessed 20 April 2018].

Gartner, 2017. *Gartner.* [Online]
Available at: https://blogs.gartner.com/smarterwithgartner/files/2017/08/Emerging-Technology-Hype-Cycle-for-2017_Infographic_R6A.jpg
[Accessed 3 November 2017].

Geer, R., 2001. *NCBI Genetics Review.* [Online]
Available at:
https://www.ncbi.nlm.nih.gov/Class/MLACourse/Original8Hour/Genetics/gene.html
[Accessed 4 May 2018].

Geng, D. & Shannon, S., 2017. *University of Berkeley's Machine Learning Blog.* [Online]
Available at: https://ml.berkeley.edu/blog/2017/02/04/tutorial-3/
[Accessed 19 April 2018].

Gibson, A., Nicholson, C. & Patterson, J., n.d. *Deeplearning4j.* [Online]
Available at: https://deeplearning4j.org/generative-adversarial-network#generative-vs-discriminative-algorithms
[Accessed 21 April 2018].

Goodfellow, I., Bengio, Y. & Courville, A., 2016. *Deep Learning (Adaptive Computation and Machine Learning).* 1. ed. Cambridge: MIT Press.

Goodfellow, I. et al., 2014. *Arxiv: Generative Adversarial Nets.* [Online]
Available at: https://arxiv.org/pdf/1406.2661.pdf
[Accessed 21 April 2018].

Google, 2018. *Google Trends.* [Online]
Available at: https://trends.google.de/trends/explore?date=2008-05-06%202018-05-06&q=%2Fm%2F0h1fn8h
[Accessed 6 May 2018].

Grönheim, M., 2018. *Interview regarding Lightning Strikes on Aircraft* [Interview] (23 March 2018).

Gridlogics, n.d. *Patent Analysis.* [Online]
Available at: http://patentanalysis.org/articles-on-patent-analysis/visual-patent-analysis/
[Accessed 29 April 2018].

Griffiths, A., 2014. *dezeen.* [Online]
Available at: https://www.dezeen.com/2014/08/05/biomimicry-3d-printed-soft-seat-chair-by-lilian-van-daal/
[Accessed 15 07 2017].

Griffith, T., 2017. *Redis Labs.* [Online]
Available at: https://redislabs.com/blog/introduction-redis-ml-part-three/
[Accessed 23 January 2018].

Guthix, S. Z., 2014. *Stackexchange Biology.* [Online]
Available at: https://biology.stackexchange.com/questions/21082/how-does-core-conductor-model-correspond-to-an-actual-neuron
[Accessed 6 May 2018].

Haggerty, A., 2004. *Massachusetts Institute of Technology.* [Online]
Available at: https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-885j-aircraft-systems-engineering-fall-2004/lecture-notes/lifecycle.pdf
[Accessed 6 April 2018].

Haley, M., 2017. *Autodesk Montreal Tech Talks: From AI for Design to AI for Making.* Montreal, Autodesk.

Hassabis, D. & Silver, D., 2017. *DeepMind's Blog.* [Online]
Available at: https://deepmind.com/blog/alphago-zero-learning-scratch/
[Accessed 16 January 2018].

Hastie, T., Tibshirani, R. & Friedman, J., 2008. *The Elements of Statistical Learning.* 2. ed. New York: Springer Science+Business Media.

Hedayati, R. & Sadighi, M., 2015. *Bird Strike: An Experimental, Theoretical and Numerical Investigation.* 1. ed. Sawston: Woodhead Publishing.

Heinrich, J. & Silver, D., 2016. *Arxiv: Deep Reinforcement Learning from Self-Play in Imperfect-Information Games.* [Online]
Available at: https://arxiv.org/pdf/1603.01121.pdf
[Accessed 22 April 2018].

Heisler, Y., 2017. *BGR.* [Online]
Available at: http://bgr.com/2017/09/26/apple-watch-sales-estimate-33-million-revenue/
[Accessed 24 March 2018].

He, K., Gkioxari, G., Dollar, P. & Girshick, R., 2018. *Arxiv.* [Online]
Available at: https://arxiv.org/pdf/1703.06870.pdf
[Accessed 17 April 2018].

He, K., Zhang, X., Ren, S. & Sun, J., 2015. *Arxiv.* [Online]
Available at: Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification
[Accessed 9 April 2018].

Helmholtz Association, n.d. *Helmholtz Association.* [Online]
Available at:
https://www.helmholtz.de/en/transfer/technology_transfer/transfer_instruments/
[Accessed 4 May 2018].

Hern, A., 2017. *The Guardian.* [Online]
Available at: https://www.theguardian.com/technology/2017/sep/04/elon-musk-ai-third-world-war-vladimir-putin
[Accessed 14 September 2017].

Hicks, B., Larsson, A., Culley, S. & Larsson, T., 2009. *A Methodology for Evaluating Technology Readiness during Product Development.* Stanford, Stanford University.

Hinton, G. E., Osindero, S. & Teh, Y.-W., 2006. A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation,* Issue 18, p. 1527–1554.

Hinton, G. et al., 2016. *Youtube: Lecture 7.5 — Long term Short term memory [Neural Networks for Machine Learning].* [Online]

Available at: https://www.youtube.com/watch?v=93rzMHtYT_0
[Accessed 20 April 2018].

Hosseinzadeh, F., Kayvanjoo, A. H., Ebrahimi, M. & Goliaei, B., 2013. Prediction of lung tumor types based on protein attributes by machine learning algorithms. *SpringerPlus,* 2(1), p. 238.

Hubel, D. H. & Wiesel, T. N., 1962. RECEPTIVE FIELDS, BINOCULAR INTERACTION AND FUNCTIONAL ARCHITECTURE IN THE CAT'S VISUAL CORTEX. *J. Physiology,* Volume 160, pp. 106-154.

Hurley, S., 2016. *Autodesk Blog.* [Online]
Available at: http://autodesk.blogs.com/between_the_lines/2016/07/what-is-generative-design.html
[Accessed 21 September 2017].

Huttenlocher, D., 2009. *Cornell University: Tutorial on Pictorial Structures for Matching and Recognition.* [Online]
Available at: http://www.cs.cornell.edu/~dph/talks/huttenlocher_icvss09.pdf
[Accessed 27 April 2018].

HV, 2017. *Null Pointer Exception.* [Online]
Available at: https://nullpointerexception1.wordpress.com/2017/12/16/a-tutorial-to-understand-decision-tree-id3-learning-algorithm/
[Accessed 30 January 2018].

Isola, P., Zhu, J.-Y., Zhou, T. & Efros, A. A., 2017. *Arxiv: Image-to-Image Translation with Conditional Adversarial Networks.* [Online]
Available at: https://arxiv.org/pdf/1611.07004.pdf
[Accessed 22 April 2018].

Ivan, M., 2017. *Proft.me.* [Online]
Available at: http://en.proft.me/2017/01/22/classification-using-k-nearest-neighbors-r/
[Accessed 30 January 2018].

Jain, R., 2017. *Hacker Earth Blog.* [Online]
Available at: http://blog.hackerearth.com/wp-content/uploads/2017/02/kernel.png
[Accessed 29 January 2018].

Janschitz, M., 2014. *t3n - digital pioneers.* [Online]
Available at: http://t3n.de/news/turing-test-ki-eugene-goostman-549886/
[Accessed 24 August 2017].

Jordan, J., 2017. *Jeremy Jordan's Blog.* [Online]
Available at: https://www.jeremyjordan.me/markov-decision-process/
[Accessed 22 April 2018].

Jurafsky, D., 2016. *Stanford Class CS214.* [Online]
Available at: https://web.stanford.edu/class/cs124/lec/languagemodeling.pdf
[Accessed 13 April 2018].

Karim, M. R., 2016. *DZone.* [Online]
Available at: https://dzone.com/articles/deep-learning-via-multilayer-perceptron-classifier
[Accessed 25 February 2018].

Karpathy, A., 2015. *Andrej Karpathy's Personal Stanford Homepage.* [Online]
Available at: https://cs.stanford.edu/people/karpathy/sfmltalk.pdf
[Accessed 19 April 2018].

Karpathy, A., 2018. *CS231n Notes on GitHub.* [Online]
Available at: http://cs231n.github.io/convolutional-networks/
[Accessed 15 April 2018].

Kendall, A., Cipolla, R. & Badrinarayanan, V., 2015. *Cambridge University.* [Online]
Available at: http://mi.eng.cam.ac.uk/projects/segnet/#demo
[Accessed 26 August 2017].

Kim, Y. G., Suh, J. H. & Park, S. C., 2008. Visualization of patent analysis for emerging technology. *Expert Systems with Applications,* Issue 34, pp. 1804-1812.

Kiser, M., 2016. *Algorithmia.* [Online]
Available at: https://blog.algorithmia.com/introduction-natural-language-processing-nlp/
[Accessed 20 February 2018].

Koehrsen, W., 2017. *Medium; Object Recognition with Google's Convolutional Neural Networks.* [Online]
Available at: https://medium.com/@williamkoehrsen/object-recognition-with-googles-convolutional-neural-networks-2fe65657ff90
[Accessed 17 April 2018].

Kouamo, S. & Tangha, C., 2012. Distributed Computing and Artificial Intelligence: 9th International Conference. In: J. F. D. P. S. S. R. G. J. M. M. A. M. B. a. J. M. C. R. Sigeru Omatu, ed. *Handwritten Character Recognition with Artificial Neural Networks.* Heidelberg: Springer, pp. 535-543.

Krizhevsky, A., Sutskever, I. & Hinton, G. E., 2012. *ImageNet Classification with Deep Convolutional Neural Networks.* [Online]
Available at: https://www.cs.toronto.edu/~fritz/absps/imagenet.pdf

Kruse, F., 2017. *Leichtbau Vorlesung im SS17.* Hamburg: HAW Hamburg.

Kuan, K. et al., 2017. *Deep Learning for Lung Cancer Detection: Tackling the Kaggle Data Science Bowl 2017 Challenge,* Singapur: Institute for Infocomm Research.

Kubota, T., 2017. *Stanford University News.* [Online]
Available at: https://news.stanford.edu/2017/01/25/artificial-intelligence-used-identify-skin-cancer/
[Accessed 15 March 2018].

Kurzweil, R., 2005. *The Singularity Is Near: When Humans Transcend Biology.* 1. ed. London: Penguin Books.

LeCun, Y., 2016. *Facebook Research.* [Online]
Available at: https://research.fb.com/artificial-intelligence-revealed/
[Accessed 17 April 2018].

LeCun, Y., Bengio, Y. & Hinton, G., 2015. Deep Learning. *Nature,* Issue 521, pp. 436-444.

LH, 2018. *Lufthansa.* [Online]
Available at: https://www.lufthansa.com/de/de/Sitzplaene_A320-200
[Accessed 7 April 2018].

LHT, 2018. *Photographs of lightning strike damages ,* Frankfurt: LHT internal.

Li, F.-F., Johnson, J. & Yeung, S., 2017. *Stanford University, CS231n.* [Online]
Available at: http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture10.pdf
[Accessed 18 April 2018].

Liker, J. K. & Meier, D. P., 2007. *Praxisbuch, der Toyota-Weg.* 1. ed. Munich: FinanzBuch Verlag.

Linick, J., 2017. *ICEAA 2017: Technology Readiness Level (TRL) vs. Percent Development Cost.* Portland, ICEAA Online.

Linstone, H. A. & Turoff, M., 2002. *New Jersey Institute of Technology: The Delphi Method Techniques and Applications.* [Online]
Available at: https://web.njit.edu/~turoff/pubs/delphibook/delphibook.pdf
[Accessed 29 April 2018].

Lin, T.-Y.et al., 2015. *Arxiv: Microsoft COCO: Common Objects in Context.* [Online]
Available at: https://arxiv.org/pdf/1405.0312.pdf
[Accessed 3 May 2018].

LISA Lab., 2018. *Deep Learning.* [Online]
Available at: http://deeplearning.net/tutorial/mlp.html
[Accessed 25 February 2018].

Long, 2017. *Medium; Convolutional Neural Network.* [Online]
Available at: https://medium.com/@Aj.Cheng/convolutional-neural-network-d9f69e473feb
[Accessed 17 April 2018].

Lozano-Pérez, T. & Kaelbling, L., 2002. *MIT Open Courseware Lecture Notes on SMA 5504.* [Online]
Available at: https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-825-techniques-in-artificial-intelligence-sma-5504-fall-2002/lecture-notes/Lecture20FinalPart1.pdf
[Accessed 22 April 2018].

Lozovschi, A., 2017. *Tech Times.* [Online]
Available at: http://www.techtimes.com/articles/200137/20170304/artificial-intelligence-its-not-man-vs-machine-say-ibm-and-google.htm
[Accessed 21 August 2017].

Mályusz, L. & Varga, A., 2017. Creative Construction Conference 2017: An Estimation of the Learning Effect on Project Cost Scheduling. *Procedia Engineering,* Issue 196, pp. 723-729.

Ma, B., Nahal, S. & Tran, F., 2016. Future Reality: Virtual, Augmented & Mixed Reality (VR, AR & MR) Primer. *Thematic Investing*, 05 September, p. 14.

Machart, P., 2018. *Talk about Functionality and Progress of Generative Models* [Interview] (19 March 2018).

Magruder, S., 2018a. *Mask-R-CNN presentation at the ZMNH / BonnLab Journal Club,* Hamburg: Institute of Medical Systems Biology.

Magruder, S., 2018b. *Development of novel graph and deep learning-based algorithms in biomedicine,* Hamburg: eigen.

Magruder, S., 2018c. *PhD Study Report,* Hamburg: ZMNH.

Magruder, S., 2018d. *Interview about the analog development effort and parameter settings from Detexon to AutoInspect* [Interview] (6 May 2018d).

Mahadevan, S., 2018. *University of Massachussets Amherst.* [Online]
Available at: https://people.cs.umass.edu/~mahadeva/papers/aaai2018-imagination.pdf
[Accessed 23 April 2018].

Majumdar, W. S., 2007. *Naval Postgraduate School: System of systems technology readiness assessment.* [Online]
Available at:
https://calhoun.nps.edu/bitstream/handle/10945/3273/07Sep_Majumdar.pdf?sequence=

1
[Accessed 4 May 2018].

Mallick, S., 2016. *Learn OpenCV.* [Online]
Available at: https://www.learnopencv.com/image-recognition-and-object-detection-part1/
[Accessed 17 April 2018].

Maney, K., 2017. *Newsweek.* [Online]
Available at: http://www.newsweek.com/2017/08/18/ai-facebook-artificial-intelligence-machine-learning-robots-robotics-646944.html
[Accessed 14 September 2017].

Mankar, V., 2011. *InTech Open.* [Online]
Available at: https://cdn.intechopen.com/pdfs-wm/18211.pdf
[Accessed 8 February 2018].

Marcus, G., 2018. *Arxiv: Deep Learning: A Critical Appraisal.* [Online]
Available at: https://arxiv.org/pdf/1801.00631.pdf
[Accessed 7 May 2018].

Marr, B., 2016. *Forbes.* [Online]
Available at: https://www.forbes.com/sites/bernardmarr/2016/09/30/what-are-the-top-10-use-cases-for-machine-learning-and-ai/
[Accessed 14 September 2017].

Matiisen, T., 2015. *Intel AI Blog.* [Online]
Available at: https://ai.intel.com/demystifying-deep-reinforcement-learning/
[Accessed 22 April 2018].

McCullock, J., n.d. *Mnemosyne Studio.* [Online]
Available at: http://mnemstudio.org/path-finding-q-learning-tutorial.htm
[Accessed 22 April 2018].

McMullan, D., 2014. *Genome.* [Online]
Available at: http://genomemag.com/what-is-personalized-medicine/
[Accessed 19 March 2018].

Metz, C., 2017. *New York Times.* [Online]
Available at: https://www.nytimes.com/2017/10/22/technology/artificial-intelligence-experts-salaries.html
[Accessed 6 May 2018].

Metz, C., 2018. *New York Times.* [Online]
Available at: https://www.nytimes.com/2018/04/19/technology/artificial-intelligence-

salaries-openai.html
[Accessed 6 May 2018].

Meyer, R., 2015. *AMAX Blog.* [Online]
Available at: https://www.amax.com/blog/?p=804
[Accessed 11 September 2017].

Miikkulainen, R. et al., 2017. *Arxiv: Evolving Deep Neural Networks.* [Online]
Available at: https://arxiv.org/pdf/1703.00548v1.pdf
[Accessed 23 April 2018].

Mitchell, P., 2017. *Spend Matters.* [Online]
Available at: http://spendmatters.com/2017/01/04/artificial-intelligence-contract-management-part-2-knowledge-representation/
[Accessed 13 April 2018].

MIT, n.d. *Moral Machine.* [Online]
Available at: http://moralmachine.mit.edu/hl/de
[Accessed 14 September 2017].

Mitra, S., 2013. *Harvard Business Review.* [Online]
Available at: https://hbr.org/2013/09/when-big-companies-support-startups?cm_sp=Article-_-Links-_-Comment
[Accessed 16 April 2018].

Mnih, V. et al., 2013. *Arxiv: Playing Atari with Deep Reinforcement Learning.* [Online]
Available at: https://arxiv.org/pdf/1312.5602.pdf
[Accessed 22 April 2018].

Mobley, R. K., 2002. *An Introduction to Predictive Maintenance.* 2 ed. Oxford: Butterworth-Heinemann.

Mogk, C., 2016. *Autodesk Research.* [Online]
Available at: https://www.autodeskresearch.com/blog/when-iot-meets-generative-design-cars
[Accessed 21 September 2017].

Moujahid, A., 2016. *Adil Moujahid's Blog.* [Online]
Available at: http://adilmoujahid.com/posts/2016/06/introduction-deep-learning-python-caffe/
[Accessed 16 January 2018].

National Instruments, 2008. *National Instruments.* [Online]
Available at: http://www.ni.com/white-paper/3470/de/
[Accessed 26 August 2017].

Ng, A., 2013. *Andrew Ng: Deep Learning, Self-Taught Learning and Unsupervised Feature Learning,* Los Angeles: University of California Los Angeles.

Ng, A., 2015. *Andrew Ng, Chief Scientist at Baidu,* San Francisco: Extract 2015 Conference.

Ng, A., 2016. *Youtube Channel of Lex Fridman: Talk at Deep Learning School.* [Online]
Available at: https://youtu.be/F1ka6a13S9I
[Accessed 9 May 2018].

Nielsen, M., 2017. *Neural Networks and Deep Learning.* [Online]
Available at: http://neuralnetworksanddeeplearning.com/chap1.html
[Accessed 4 September 2017].

Numenta, 2011. *Numenta.* [Online]
Available at: https://numenta.org/resources/HTM_CorticalLearningAlgorithms.pdf
[Accessed 23 April 2018].

Ohnsman, A., 2016. *Forbes.* [Online]
Available at: https://www.forbes.com/sites/alanohnsman/2016/10/03/toyotas-robot-car-line-in-the-sand-8-8-billion-test-miles-to-ensure-safety/
[Accessed 28 August 2017].

Okoli, C. & Pawlowski, S. D., 2004. *Science Direct: The Delphi method as a research tool: an example, design considerations and applications.* [Online]
Available at: https://ac.els-cdn.com/S0378720603001794/1-s2.0-S0378720603001794-main.pdf?_tid=44a2cf3c-8daa-4ccf-9b38-4c5013d2f111&acdnat=1525006648_c1e9b7d10541b652ae77b9d120b07208
[Accessed 29 April 2018].

Olah, C., 2015. *Colah's Blog: Understanding LSTM Networks.* [Online]
Available at: http://colah.github.io/posts/2015-08-Understanding-LSTMs/
[Accessed 20 April 2018].

Olechowski, A., Eppinger, S. D. & Joglekar, N., 2015. *Portland International Conference on Management of Engineering and Technology (PICMET): Technology readiness levels at 40: A study of state-of-the- art use, challenges, and opportunities.* Portland, MIT OpenAccess Articles.

Oppermann, A., 2018. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/deep-learning-meets-physics-restricted-boltzmann-machines-part-i-6df5c4918c15
[Accessed 8 May 2018].

Ouaknine, A., 2018. *Medium: Review of Deep Learning Algorithms for Object Detection.* [Online]
Available at: https://medium.com/comet-app/review-of-deep-learning-algorithms-for-object-detection-c1f3d437b852
[Accessed 3 May 2018].

Pahl, G., Beitz, W., Feldhusen, J. & Grote, K.-H., 2006. *Konstruktionslehre: Grundlagen erfolgreicher Produktentwicklung. Methoden und Anwendung.* 7. ed. Berlin: Springer Verlag.

Panetta, K., 2017. *Gartner.* [Online]
Available at: https://www.gartner.com/smarterwithgartner/top-trends-in-the-gartner-hype-cycle-for-emerging-technologies-2017/
[Accessed 3 November 2017].

Parisi, G. I. et al., 2018. *Arxiv: Continual Lifelong Learning with Neural Networks: A Review.* [Online]
Available at: https://arxiv.org/pdf/1802.07569.pdf
[Accessed 2 May 2018].

Parziale, G., 2016. *Invasive Code.* [Online]
Available at: https://www.invasivecode.com/weblog/convolutional-neural-networks-ios-10-macos-sierra/
[Accessed 17 April 2018].

Peeling, P., 2017. *Matlab Expo.* [Online]
Available at: http://www.matlabexpo.com/uk/2017/proceedings/big-data-machine-learning-for-predictive-maintenance.pdf
[Accessed 8 April 2018].

Peng, X. B., Berseth, G., Yin, K. & Van de Panne, M., 2017. *University of British Columbia: DeepLoco: Dynamic Locomotion Skills Using Hierarchical Deep Reinforcement Learning.* [Online]
Available at: http://www.cs.ubc.ca/~van/papers/2017-TOG-deepLoco/2017-TOG-deepLoco.pdf
[Accessed 22 April 2018].

Perez, C. E., 2016. *The Alien Style of Deep Learnign Generative Design.* [Online]
Available at: https://medium.com/intuitionmachine/the-alien-look-of-deep-learning-generative-design-5c5f871f7d10
[Accessed 13 07 2017].

Pierobon, M., 2015. *Aviation Pros.* [Online]
Available at: http://www.aviationpros.com/article/12076178/aircraft-ground-damage-

reporting
[Accessed 9 April 2018].

Poplin, R. et al., 2018. Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning. *Nature Biomedical Engineering,* 2(3), pp. 158-164.

Protalinski, E., 2017. *Venture Beat.* [Online]
Available at: https://venturebeat.com/2017/05/17/googles-speech-recognition-technology-now-has-a-4-9-word-error-rate/
[Accessed 24 September 2017].

Radhakrishnan, P., 2017a. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/image-captioning-in-deep-learning-9cd23fb4d8d2
[Accessed 19 April 2018].

Radhakrishnan, P., 2017b. *Towards Data Science: Hyperparameters.* [Online]
Available at: https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a
[Accessed 2 May 2018].

Ragate, S. & Asai, R., 2017. *Colfax Research: OPTIMIZATION OF REAL-TIME OBJECT DETECTION ON INTEL® XEON® SCALABLE PROCESSORS.* [Online]
Available at: https://colfaxresearch.com/download/7136/
[Accessed 5 May 2018].

Rajagopal, J., 2015. *Youtube: Recurrent Neural Networks - Ep. 9 (Deep Learning SIMPLIFIED).* [Online]
Available at: https://www.youtube.com/watch?v=_aCuOwF1ZjU&t=186s
[Accessed 20 April 2018].

Ramachandra, G., 2017. *Hackernoon.* [Online]
Available at: https://hackernoon.com/generative-adversarial-networks-a-deep-learning-architecture-4253b6d12347
[Accessed 22 April 2018].

Raschka, S., 2018. *MLxtend Documentation GitHub.* [Online]
Available at: https://rasbt.github.io/mlxtend/user_guide/general_concepts/gradient-optimization/
[Accessed 24 January 2018].

Ravi, S. & Larochelle, H., 2017. *OpenReview: Optimization as a Model for Few-Shot Learning.* [Online]
Available at: https://openreview.net/pdf?id=rJY0-Kcll
[Accessed 23 April 2018].

Redmon, J., Divvala, S., Girshick, R. & Farhadi, A., 2015. *PJ Reddie.* [Online]
Available at: https://pjreddie.com/media/files/papers/yolo.pdf
[Accessed 17 April 2018].

Redmon, J. & Farhadi, A., 2016. *Arxiv: YOLO9000: Better, Faster, Stronger.* [Online]
Available at: https://arxiv.org/pdf/1612.08242.pdf
[Accessed 3 May 2018].

Reid, S., 2014. *Turing Finance.* [Online]
Available at: http://www.turingfinance.com/misconceptions-about-neural-networks/
[Accessed 14 April 2018].

Reusch, B., 1993. *Fuzzy Logic: Theorie und Praxis.* Berlin, Springer Verlag.

Reuters, 2007. *Reuters.* [Online]
Available at: https://www.reuters.com/article/us-india-summit-outsourcing/factbox-key-facts-on-indias-outsourcing-sector-idUSSP4999820060207
[Accessed 16 Dezember 2017].

Riedmiller, M., 2010. *Universität Freiburg.* [Online]
Available at: http://ml.informatik.uni-freiburg.de/former/_media/teaching/ss10/05_mlps.printer.pdf
[Accessed 24 February 2018].

Rieke, J., 2017. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/object-detection-with-neural-networks-a4e2c46b4491
[Accessed 3 May 2018].

Ries, E., 2015. *Lean Startup: Schnell, risikolos und erfolgreich Unternehmen gründen.* 4. ed. München: Redline Verlag.

Rindt, B., 2018. *Interview with a maintenance engineer at Lufthansa Technik Base Frankfurt* [Interview] (18 April 2018).

Rosebrock, A., 2016. *PyImageResearch.* [Online]
Available at: https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/
[Accessed 3 May 2018].

Ruder, S., 2017. *Arxiv: An overview of gradient descent optimization algorithms.* [Online]
Available at: https://arxiv.org/pdf/1609.04747.pdf
[Accessed 9 May 2018].

Russakovsky, O. et al., 2015. *Arxiv: ImageNet Large Scale Visual Recognition Challenge.* [Online]
Available at: https://arxiv.org/abs/1409.0575
[Accessed 3 May 2018].

Russell, S. J. & Norvig, P., 1995. *Artificial Intelligence: A Modern Approach.* 1 ed. New Jeesey: Prentice Hall Verlag.

RWTH Aachen, n.d. *RWTH Aachen Fakultät Informatik.* [Online]
Available at: https://www-i6.informatik.rwth-aachen.de/web/Research/speech_recog.html
[Accessed 26 August 2017].

Santo, D., 2016. *EE Times.* [Online]
Available at: https://www.eetimes.com/author.asp?section_id=36&doc_id=1330069
[Accessed 6 May 2018].

Sarmah, D., 2014. *SlideShare.* [Online]
Available at: https://de.slideshare.net/deepaneeta/overview-of-drug-toxicity
[Accessed 18 March 2018].

Sawada, Y. et al., 2017. *Arxiv: All-Transfer Learning for Deep Neural Networks and its Application to Sepsis Classification.* [Online]
Available at: https://arxiv.org/pdf/1711.04450.pdf
[Accessed 8 May 2018].

Schapire, R., 2008. *Princeton University.* [Online]
Available at:
https://www.cs.princeton.edu/courses/archive/spr08/cos511/scribe_notes/0204.pdf
[Accessed 26 August 2017].

Schröer, T., 2017. *IBM THINK Blog.* [Online]
Available at: https://www.ibm.com/de-de/blogs/think/2017/05/24/watson-qualitatskontrolle/
[Accessed 10 April 2018].

Schroeder, W. et al., 2009. *Universität Oldenburg - Gruppe "Künstliche Intelligenz".* [Online]
Available at: http://www.informatik.uni-oldenburg.de/~iug08/ki/Grundlagen_Arten_von_KI.html
[Accessed 26 August 2017].

Schuld, M., Sinayskiy, I. & Petruccione, F., 2014. *Arxiv: The Quest for a Quantum Neural Network.* [Online]

Available at: https://arxiv.org/pdf/1408.7005.pdf
[Accessed 24 April 2018].

Sergeev, A. & Del Balso, M., 2017. *Uber.* [Online]
Available at: https://eng.uber.com/horovod/
[Accessed 8 May 2018].

Serrano, L., 2016a. *A Friendly Introduction to Machine Learning,* Silicon Valley: Udacity.

Serrano, L., 2016b. *A friendly introduction to Deep Learning and Neural Networks,* Silicon Valley: Udacity.

Serrano, L., 2017a. *A Friendly Introduction To Convolutional Neural Networks and Image Recognition.* [Online]
Available at: https://www.youtube.com/watch?v=2-Ol7ZB0MmU
[Accessed 17 April 2018].

Serrano, L., 2017b. *A Friendly Introduction to Recurrent Neural Networks.* [Online]
Available at: https://www.youtube.com/watch?v=UNmqTiOnRfg
[Accessed 18 April 2018].

Seville Report, 2017. *Medium: Becoming Human.* [Online]
Available at: https://becominghuman.ai/where-to-find-artificial-intelligence-and-machine-learning-profits-in-the-market-11ebcc5637dc
[Accessed 8 May 2018].

Shafiee, M. J., Chywl, B., Li, F. & Wong, A., 2017. *Universtiy of Waterloo: Fast YOLO: A Fast You Only Look Once System for Real-time Embedded Object Detection in Video.* [Online]
Available at: http://openjournals.uwaterloo.ca/index.php/vsl/article/viewFile/171/141
[Accessed 5 May 2018].

Shaikh, F., 2017. *Analytics Vidhya.* [Online]
Available at: https://www.analyticsvidhya.com/blog/2017/04/comparison-between-deep-learning-machine-learning/
[Accessed 3 February 2018].

Sharma, A., 2015. *Digg Data.* [Online]
Available at: http://diggdata.in/post/94066544971/support-vector-machine-without-tears
[Accessed 27 January 2018].

Sharma, G., 2016. *Chatbots Magazine.* [Online]
Available at: https://chatbotsmagazine.com/artificial-intelligence-ai-and-fintech-part-1-7cae1e67dc13
[Accessed 24 September 2017].

Shibuya, N., 2017. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/understanding-generative-adversarial-networks-4dafc963f2ef
[Accessed 22 April 2018].

Silver, D., 2016. *DeepMind.* [Online]
Available at: https://deepmind.com/blog/deep-reinforcement-learning/
[Accessed 22 April 2018].

Sinfield, J. V., Gustafson, T. & Hindo, B., 2014. The Discipline of Creativity. *MIT Sloan Management Review,* Issue Winter.

Singh, M., 2017. *Udemy.* [Online]
Available at: https://www.udemy.com/machine-learning-fun-and-easy-using-python-and-keras/?couponCode=YOUTUBE_ML
[Accessed 30 January 2018].

Skin Cancer Foundation, 2018. *Skin Cancer Foundation.* [Online]
Available at: https://www.skincancer.org/skin-cancer-information/skin-cancer-facts
[Accessed 15 March 2018].

Smith, A., 2016. *Odyssey.* [Online]
Available at: https://www.theodysseyonline.com/artificial-intelligence-from-myth-to-reality
[Accessed 14 August 2017].

Smith, L., 2002. *University of Otago, New Zealand.* [Online]
Available at:
http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf
[Accessed 31 January 2018].

Stövesand, J., 2017. *Interview for the Use of Deep Learning in Aviation Applications* [Interview] (11 December 2017).

Statista, 2017. *Statista.* [Online]
Available at: https://www.statista.com/statistics/607835/worldwide-artificial-intelligence-market-leading-use-cases/
[Accessed 8 March 2018].

Stegagno, P., 2018. *Max-Planck-Institut für biologische Kybernetik.* [Online]
Available at: http://www.kyb.tuebingen.mpg.de/de/forschung/abt/bu/hri/indoor-localization-of-uavs.html
[Accessed 6 May 2018].

Tally, S., 2012. *Purdue University.* [Online]
Available at:

http://www.purdue.edu/newsroom/research/2012/120219KlimeckAtom.html
[Accessed 14 September 2017].

Tanaka, H., 2016. *Deep Learning for Advanced 3D Printing.* Irvine, National Academy of Engineering.

Tanaka, H., 2017. *"Alchemist's Canvas" (Deep Learning for sketch-based 3D modelling),* s.l.: s.n.

Taylor, M. E., Kuhlmann, G. & Stone, P., 2008. Transfer Learning and Intelligence: an Argument and Approach. *FRONTIERS IN ARTIFICIAL INTELLIGENCE AND APPLICATIONS,* 171(326).

Techopedia, n.d. *Techopedia.* [Online]
Available at: https://www.techopedia.com/definition/190/artificial-intelligence-ai
[Accessed 6 August 2017].

Thangaratinam, S. & Redman, C. W., 2005. The Delphi technique. *The Obstetrician & Gynaecologist,* Issue 7, pp. 120-125.

Tommasi, T., Patricia, N., Caputo, B. & Tuytelaars, T., 2015. *Arxiv: A Deeper Look at Dataset Bias.* [Online]
Available at: https://arxiv.org/pdf/1505.01257.pdf
[Accessed 8 May 2018].

Toren, A., 2016. *Entrepreneur.* [Online]
Available at: https://www.entrepreneur.com/article/277349
[Accessed 5 April 2018].

Trevino, A., 2016. *Data Science.* [Online]
Available at: https://www.datascience.com/blog/k-means-clustering
[Accessed 1 February 2018].

UE Systems, 2014. *UE Systems.* [Online]
Available at: http://www.uesystems.com/mechanical-inspection/the-4-basic-maintenance-modes
[Accessed 8 April 2018].

Uijlings, J., van de Sande, K., Gevers, T. & Smeulders, A., 2012. *Huppelen.* [Online]
Available at: http://www.huppelen.nl/publications/selectiveSearchDraft.pdf
[Accessed 3 May 2018].

Uz, F. B., 2017. *Microsoft Azure Blog: Internet of Things.* [Online]
Available at: https://azure.microsoft.com/de-de/blog/deep-learning-for-predictive-maintenance/
[Accessed 8 April 2018].

Valich, T., 2015. *VR World.* [Online]
Available at: https://vrworld.com/2015/05/08/big-data-in-planes-new-pw-gtf-engine-telemetry-to-generate-10gbs/
[Accessed 7 Juli 2017].

Van der Velde, N., 2018. *Global Me.* [Online]
Available at: https://www.globalme.net/blog/how-does-speech-recognition-technology-work
[Accessed 21 February 2018].

Van Norman, G. A. & Eisenkot, R., 2017. Technology Transfer: From the Research Bench to Commercialization: Part 2: The Commercialization Process. *JACC: Basic to Translational Science,* 2(2), pp. 197-208.

Vincent, J., 2018. *The Verge.* [Online]
Available at: https://www.theverge.com/2018/2/19/17027902/google-verily-ai-algorithm-eye-scan-heart-disease-cardiovascular-risk
[Accessed 13 March 2018].

Voigt, C., 2017. *Presseportal.* [Online]
Available at: http://www.presseportal.de/pm/125448/3575790
[Accessed 4 August 2017].

Vreeken, J., 2003. *Jonad: Spiking neural networks, an introduction.* [Online]
Available at: http://www.ai.jonad.eu/materialy/download/sieci_neuronowe/2003-008.pdf
[Accessed 23 April 2018].

Vullings, R., 2013. *SlideShare.* [Online]
Available at: https://de.slideshare.net/ramonvullings/27-creativity-innovation-tools-final
[Accessed 30 April 2018].

Walker, M. J., 2016. *Forbes.* [Online]
Available at: https://www.forbes.com/sites/gartnergroup/2016/08/29/track-three-trends-in-the-2016-gartner-hype-cycle-for-emerging-technologies/#1c165e165e6c
[Accessed 4 August 2017].

Wang, H. & Raj, B., 2017. *Arxiv: On the Original of Deep Learning.* [Online]
Available at: https://arxiv.org/pdf/1702.07800.pdf
[Accessed 23 April 2018].

Wang, T., Chen, Y., Qiao, M. & Snoussi, H., 2017. *Springer.* [Online]
Available at: https://link.springer.com/content/pdf/10.1007%2Fs00170-017-0882-0.pdf
[Accessed 11 April 2018].

Warth, D., 1999. *Universität Mainz.* [Online]
Available at: http://www.fask.uni-mainz.de/user/warth/Ki.html#Heading6
[Accessed 26 August 2017].

Weng, L., 2017. *Lil'Log.* [Online]
Available at: https://lilianweng.github.io/lil-log/2017/06/21/an-overview-of-deep-learning.html
[Accessed 9 May 2018].

WHO, 2017. *World Health Organization.* [Online]
Available at: http://www.who.int/mediacentre/factsheets/fs282/en/
[Accessed 21 March 2018].

WHO, 2018. *World Health Organization.* [Online]
Available at: http://www.who.int/mediacentre/factsheets/fs339/en/
[Accessed 5 April 2018].

Winterhoff, M. et al., 2009. *Arthur D Little.* [Online]
Available at:
http://www.adlittle.de/uploads/tx_extthoughtleadership/ADL_Zukunft_der_Mobilitaet_2020_Langfassung.pdf
[Accessed 5 Juli 2017].

WIPO, 2018. *World Intellectual Property Organization.* [Online]
Available at: www3.wipo.int
[Accessed 29 April 2018].

Yakovenko, N., 2018. *Medium.* [Online]
Available at: https://medium.com/@Moscow25/the-ai-talent-shortage-704d8cf0c4cc
[Accessed 6 May 2018].

YCharts, 2018. *Y Charts.* [Online]
Available at: https://ycharts.com/companies/AAPL/r_and_d_expense
[Accessed 8 May 2018].

Youtube, 2017. *Youtube.* [Online]
Available at: https://www.youtube.com/watch?v=o-MlJI48XX4
[Accessed 10 April 2018].

Zachevsky, I., 2012. *Technion.* [Online]
Available at:
http://lihi.eew.technion.ac.il/files/Teaching/2012_winter_048921/PPT/Ido.pdf
[Accessed 12 April 2018].

Zhou, F., Wu, B. & Li, Z., 2018. *Arxiv: Deep Meta-Learning: Learning to Learn in the Concept Space.* [Online]

Available at: https://arxiv.org/pdf/1802.03596.pdf
[Accessed 23 April 2018].

Zhuang, Y., Cappos, J., Rappaport, T. S. & McGeer, R., 2013. *Polytechnic Institute of New York University.* [Online]
Available at: https://ssl.engineering.nyu.edu/papers/tr-cse-2013-04.pdf
[Accessed 7 Juli 2017].

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Gemäß der Allgemeinen Prüfungs- und Studienordnung ist zusammen mit der Abschlussarbeit eine schriftliche Erklärung abzugeben, in der der Studierende bestätigt, dass die Abschlussarbeit „– bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit [(§ 18 Abs. 1 APSO-TI-BM bzw. § 21 Abs. 1 APSO-INGI)] – ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt wurden. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich zu machen.“

Quelle: § 16 Abs. 5 APSO-TI-BM bzw. § 15 Abs. 6 APSO-INGI

Dieses Blatt, mit der folgenden Erklärung, ist nach Fertigstellung der Abschlussarbeit durch den Studierenden auszufüllen und jeweils mit Originalunterschrift als <u>letztes Blatt</u> in das Prüfungsexemplar der Abschlussarbeit einzubinden.
Eine unrichtig abgegebene Erklärung kann -auch nachträglich- zur Ungültigkeit des Studienabschlusses führen.

Erklärung zur selbstständigen Bearbeitung der Arbeit

Hiermit versichere ich,

Name: ______________________________

Vorname: ______________________________

dass ich die vorliegende – bzw. bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit – mit dem Thema:

ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

- die folgende Aussage ist bei Gruppenarbeiten auszufüllen und entfällt bei Einzelarbeiten -

Die Kennzeichnung der von mir erstellten und verantworteten Teile der ist erfolgt durch:

Ort	Datum	Unterschrift im Original