Ram Krishna Banstola

Applications of Images Anomalies Detection using Deep Learning in Department Store

Master's thesis of Faculty of Information Technologies

December 8, 2020

University of Jyväskylä

Faculty of Information Technology

Author: Ram Krishna Banstola

Contact information: ram.banstola@gmail.com

Supervisors: Dr. Oleksiy Khriyenko

Title: Applications of Images Anomalies Detection using Deep Learning in Department Store

Project: Master's thesis

Study line: Web Intelligence and Service Engineering

Page count: 65

Abstract:

Deep learning is a branch of machine learning which itself is a branch of Artificial Intelligence. The use of deep learning to solve domain specific problems is on the rise. Deep learning has been successfully used to assist sales prediction in retail, disease detection in medicine, road infrastructure monitoring by checking cracks on the road, accidents prone zones, detect anomalous activities in the realm of cyber security etc. At present, user and machine generated data is available abundantly and the challenges for enterprises is to infer new information from the available data to increase profit for the enterprise, produce a reliable system and increase customer satisfaction. Deep learning has been successfully used in classification of data with high precision. However, there are bottlenecks when it comes to anomalies in data because building models to detect anomalies is more difficult than classification problems. This thesis aims to study image anomalies detection and their applications department store using design science research methods. This thesis presents a basic prototype application to demonstrate anomalies in product areas in department stores.

Keywords: deep learning, computer vision, anomaly detection, autoencoder, retail robot

Preface

Tamasō mā jyōtirgamaya (From darkness leads us to light) is a famous prayer from Brihadaranyaka Upanishad, a Hindu scripture. This prayer guides me in the most difficult moments telling me that I must keep a positive outlook and work towards light from whatever darkness that exists. The prayer also implies from ignorance to knowledge. When writing this thesis, I got an opportunity to learn about new technologies in deep learning and how those are shaping our society. When the thesis was completed it made me more curious and I had more questions than when I started. During the writing process our daughter Aruna was born and when I held her first time, I had range of emotions and one thought that was constant was what kind of world it would be when she will grow up and what could I do so that she gets the best opportunities to succeed in life and find happiness in her pursuits in age when decisions will likely be influenced more than ever by algorithms.

I would like to thank Dr. Oleksiy Khriyenko, my thesis supervisor, from the bottom of my heart for his continuous guidance and support when writing this thesis. I feel truly blessed to have found such a humble and knowledgeable person as my thesis supervisor.

Finally, I would like to thank my parents who never got an opportunity to go to school but made sure that I get the best education available. I would like to thank my wife for her support to help me to complete the thesis even in the most difficult days after childbirth.

Helsinki, December 8, 2020

Ram Krishna Banstola

Glossary

AE AutoEncoder

AI Artificial Intelligence

API Application Programming Interface

BP Backward Propagation

CNN Convolutional Neural Network

DL Deep Learning

GAN Generative Adversarial Network

KDE Kernel Density Estimation

ML Machine Learning

MSE Mean Square Error

NLP Natural Language Processing

RNN Recurrent Neural Network

UML Unsupervised Machine Learning

List of Figures

List of Tables

Contents

1 Introduction

Artificial Intelligence (AI) applications have surged in the past decade due to abundance of data and increased computing power. Current rise in the applications of AI in education, medicine discovery, fighting cybercrime, warfare, product, and service enrichment etc. has helped to bring awareness on the capabilities of AI. AI is not a new field, first work on AI was done by Alan Turing in 1950 in his famous paper on Computing Machinery and Intelligence. The key question was *"Can machine think?"*, to which Alan purposed that the right question to ask would be *Can machine play an imitation game?* (Turing 1950). Thus, he devised now famous Turing Test which performs test on a machine based on its ability to think like a human being. Similarly, knowledge-based system emerged in the early 90s and there was an increased amount of investment in the field of AI. The interest in the field soon vanished due to lack of concrete applications and funding for new researches. Francois Chollet, author of Keras (open-source library that provides a Python interface for artificial neural networks), calls this first winter in AI. It is often portrayed in the news and magazine that AI will solve all our future problems with the promise of intelligent agents that are capable of writing computer programs and having emotions like human beings. However current research has shown that we are far from such futuristic claims(Chollet 2018).

Advancement in deep learning has been mostly in the field of pattern recognition and classification. However, if we go back in history there were certain milestone that have been important to this field; a major milestone was when Deep Blue by IBM, a chess playing computer, won against world chess champion on May 11, 1997 after a six-game match. Fast forward in 2011, IBM Watson won game called *Jeopardy!* which required complex reasoning capabilities and understanding natural language. The difference between Deep Blue and Watson is that IBM Watson had Natural Language Processing (NLP) capabilities and was able to reason. (IBM100 - Deep Blue 2020). Similarly, AlphaZero program by DeepMind (Google's research laboratory) can play *chess*, *shogi* and *go* by going through training process and learning. The *self-play* ability of AlphaZero allows it to master complex games like *go* (Wiggers 2018). Despite using brute force

approach in Deep Blue, it was able to give hope for future where machine would be able to think and reason and pave path for future research. The result are systems like IBM Watson and AlphaZero.

Current practitioner of AI refrain from calling Deep Blue an AI because of current advancement in machine learning and deep learning which focuses on generic solution instead of hard coded algorithm that can perform predefined task such as playing chess using complex algorithm. The field of AI was stagnant for almost two decades then with increase in processing power of computer following Moore's Law which states that computing power doubles every two years (Moore's Law 2020). There has been growing interest and successful experimentations and applications in the field of AI. Important aspect of such momentum is AI is democratized and readily accessible to public. AI is more accessible, and it takes few clicks to setup hardware and software for running AI experimentation. There has been surge in Deep learning research due to availability of open source project such as Keras, TensorFlow and large dataset due to increase in Internet users who generate image, video, and text data in large quantity every day. There has been growing demand for data warehouses to store data both generated by machines and humans. It is easy to setup a deep learning experiments and observe results for researchers at present than it was a decade ago (F. Chollet, Deep learning with Python 2018).

Human beings rely on a lot knowledge about the environment and intuition to function in everyday life. Contrary to human being such knowledge about environment and context should be fed to a computer as input data. Earlier attempt to AI was using knowledge base with the hope that if enough knowledge is provided to a machine then it would be able to make connection between different context and able to reason like a human being. Such knowledge base approach did not materialize. One of the well-known projects was Cyc which was an interface engine and consisted of database of statement in CycL (Cyc Language). The problem with such approach was it was difficult to devise formal rules of the informal world we live in. Such difficulties were soon realized and such knowledge base system relying on hard-coded statements were abandoned and new approach of system that were able to extract knowledge themselves were developed known as Machine Learning (Goodfellow, Bengio and Courville 2016).

In case of Finland, there has been initiatives like Elements of AI by University of Helsinki that has been successful in educating approximately 460000 people about AI, what it means and its future. The course was received well, and it is being translated into all European languages with the help of Finnish Ministry of Employment with the aim to train one percent of European Union citizen in the basics of AI. (Finland to invest in the future skills of Europeans – training one per cent of EU citizens in the basics of AI 2019).

At present AI and Deep Learning (DL) has given rise to new industries and profession. Important applications of Deep Learning are listed below:

- Natural Language Processing (NLP)

  The ability of a machine to read, comprehend and procure meaning from human languages is called Natural Language Processing (Yse 2019). NLP includes processing unstructured data generated by users on different platform such as social media, speeches etc. NLP allows machine translation which can translate one language into another, automatic speech recognition etc. Other important applications of NLP are sentiment analysis, chatbots and virtual assistants, text classification, text extraction, auto correction and speech recognition (Wolff 2020). There are already commercial solutions available such as *Watson NLP*[1], *Google Cloud Natural Language*[2], *Amazon Comprehend*[3] for NLP. Such commercial solutions provide researchers and developers to combine NLP in other application domain.

- Recommendation system and search engines

  Recommendation system and search engines have been successfully using Deep Learning for information retrieval to leverage user experience. This is possible due to available of massive contextual data. Search engines can accurately predict the behavior of a certain user as well as guide them towards what they are likely to investigate. Recommendation system helps user choose product that interest them most based on their previous browsing behavior. Such systems are useful to reduce

[1] https://www.ibm.com/cloud/watson-natural-language-understanding
[2] https://cloud.google.com/natural-language/docs
[3] https://aws.amazon.com/comprehend/

over-choice for example in e-commerce system where user have wide range of choices (Shuai, et al. 2019).

- Behavior modelling

  Human behavior prediction plays an important role in social science and healthcare. For example, in healthcare industry it is important to predict behavior for executing timely intervention. Future behavior prediction includes when a person is prone to performing an activity such as drink coffee, visit certain websites, go for exercise. Such models can use social network as the input and produce a predictive score which can be used to infer information based on a score i.e. higher the score, the likelihood to perform certain behavior is high (Phan, et al. 2016)

- Event prediction

  Event is anything that is time and location bound such as election result announcement, vaccine for a new virus is found, rise in stock market etc. Event can be related as well. For example, during 2020 US presidential election US stock market rose soon after Joe Biden was announced president-elect. Similarly, stock market rose on the news of successful testing of Corona virus (Brown 2020). Neil Bohr in 1970 famously said *"Prediction is very difficult, especially if it's about the future",* but Deep learning models can be used to predict effect of one event into another when they are related. Historical context can be used as input to assist the model to make accurate predictions. Related techniques and big data are useful for event prediction with Deep Learning (Zhao 2007).

- Image recognition, classification, and video analysis

  Image recognition and classification is amongst the most widely used application of Deep learning. Application domain are visual search engines that use photo to find similar photos, find an image based on partial image, social media platform such as Facebook, twitter, LinkedIn use to identify harmful content, provide smooth user experience by recognizing persons in a photo, smart technologies such as self-driving, industrial and household robots use deep learning to navigate the

environment to perform their task. Video analysis can be performed using deep learning, for example Clarifai is a startup company that provides cloud-based solution for understanding content in a video (Zhai 2016). YouTube removes objectionable content and performs moderation using Deep learning. About 76% of the content is automatically flagged by the Deep learning-based classifiers (Marr 2019). There are commercial providers that uses AI in the background and provides developers with API to submit video and get video analysis such as *IBM Video Analytics*[4]*, Amazon Rekognition Video*[5]. Presence of such cloud-based provider has made it easier to integrate video analysis in other domain for example to detect if explicit content video is posted in a forum website etc.

- Audio, video, and image generation

  Autoencoder based on deep learning have been successfully used to generate images, videos, and sound. Experimentation using Generative adversarial network (GAN) have shown that it is possible to generate image, video, and sound after training models on large amount of unlabeled data. The video length was not long but model was able to learn useful feature (Vondrick, Pirsiavash and Torralba 2016).

[4] https://www.ibm.com/products/video-analytics
[5] https://aws.amazon.com/rekognition/video-features/

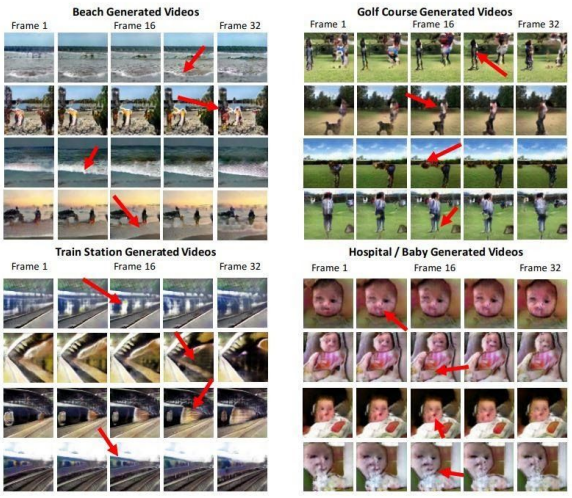

Figure 1 Video generation using GAN. Red arrow points to the direction of motion (Vondrick, Pirsiavash and Torralba 2016)

As shown in Figure 1, it is possible to generate video using GAN, a Deep Learning model. In case of child in hospital important feature such as eyes, nose and mouth in face were generated and for other scenic environmental features such as sky, ground and surrounding were generated. Similar experiments have been done with images and sound.

Such development has given rise to new threats particularly deep fake. Deep fake is a technology that allows covering face image of a victim to a video of a person in the video. This allows creating video of a victim replacing actual person in the video. Generative models are s solely used for creating deep fakes. Such activities are threating democracy, integrity of a person and creating chaos in the society because deep fakes are highly convincing, and identification can take time. Deep fake is among the most pervasive techniques used to spread fake news and propaganda. Such applications downplay the benefits of deep learning as it poses threat to national security and democracy itself. (Nguyen, et al. 2020).

Image anomalies detection is among the use cases of deep learning that can be used to solve domain specific problems in infrastructure such as road condition monitoring, medicine, cyber security, monitoring building by determining if crack in a building is anomalous or not.

Department store which must deal with arranging misplaced items in shelf on everyday basis, monitor stock on a live, monitor dangerous situation such as unusual stacking of goods and customer intentionally or unintentionally damaging the product can benefit from using image anomaly detection. When an anomalous item is detected, correct location for such item can be suggested too. Similarly, other use cases in the department store can include if customer behavior is anomalous or not by examining the most common routes takes by customers.

1.1 Research Questions and Objective

This thesis aims to answer following research questions regarding the applications of images anomalies in department store:

1. *What is the image-based anomaly detection methods?*
2. *What is the effectiveness of images anomalies detection in department store using Autoencoders and Kernel Density Estimation?*

The objective of the thesis research is to study available techniques for image anomaly detection using deep learning, study image anomalies detection as an application of deep learning and present the process of prototyping Autoencoder based model to detect images anomalies in a department store. Publications in the fields of artificial intelligence, deep learning and anomalies detection are used to study the concepts and use cases that are valid for answering the research questions.

1.2 Research Method

Design science research method (DSR) is selected for this thesis research. Design science method provides with the process to design technology assisted solution to address real-world problems that are faced by enterprises. The need for research on the new technology and design comes from daily interaction with the existing systems that could be improved by applying new knowledge and devising new system or application (Henver, et al. 2004).

The motivation for the research work arose from everyday grocery shopping experience where I saw items were being misplaced. For example, a bag of spaghetti was in bread section, fruits were placed in cold drink section etc. It was concluded that deep learning based solution can be applied to detect images anomalies in product areas in department store. It would save money and time or make the work of identifying misplaced goods easy at the department stores.

Robots are increasingly used at department and retail stores which has given rise to new industry namely Robotics-As-A-Service (RaaS). Robots provide employee assistance helping employees with fulfillment of online orders, auditing of the stocks by deploying robots to navigate various aisles in department store periodically to record items and alert central system about low stock. Robot can also perform the task of finding the item and automatically fill the aisles without human intervention. On the analytics side robots have been used to gather insight on product performance and placement. Similarly, surveillance robot that have capability to detect suspicious activity can be used to prevent possible theft (Joshi and Kumar 2014). With such existing capabilities of robots performing automated data collection; product misplacement detection could be embedded into those robots which would enhance the service provided by such robots.

Both images anomalies detection and image classification has been used widely in different sectors and there is abundance of research on how to improve both. However, there has not been any significant researches on how images anomalies detection could be used in a department store to detect and locate misplaced item and find correct location for it. Department store in a dynamic environment where artifacts have possibility to change over time and creating a seamless solution requires large image dataset from a single department store.

Department stores usually provides employee with a smart phone for communication needs and using the same device if they would be able to scan an item that is misplaced and find the correct location for the item then it would increase efficiency and decrease operational costs.

The prototype to test the concept will be developed and deployed in Google Cloud. Keras framework will be used for developing training and testing the models. Data collection will be done by capturing images from department store using a smartphone. Collected data will be annotated by hand.

Table 1 below shows research guidelines that should be followed when conducting Design-Science Research. This research method provides with processes that are useful when developing a new prototype based on current knowledge and methods.

Table 1 Design-Science Research Guidelines

Table 1. Design-Science Research Guidelines	
Guideline	**Description**
Guideline 1: Design as an Artifact	Design-science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation.
Guideline 2: Problem Relevance	The objective of design-science research is to develop technology-based solutions to important and relevant business problems.
Guideline 3: Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.
Guideline 4: Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.
Guideline 5: Research Rigor	Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.
Guideline 6: Design as a Search Process	The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.
Guideline 7: Communication of Research	Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

1.3 Thesis Outline

This thesis is divided into seven chapters; chapter one provides introduction to the thesis. Chapter two discusses theoretical aspects of deep learning and images anomalies. Chapter three discusses different type of anomalies in data and how those are relevant in

department store. Chapter four demonstrates experiment plan and illustrates how different components are combined including data collection and experiment execution. Chapter five presents the results of the experiment. Chapter six discusses the results obtained from experiment. Chapter seven concludes the thesis and presents the limitation of this thesis work and future research ideas.

2 Literature review: Deep learning

The word “neural” typically makes one to think of the brain. Data in human brain is stored using neuron which is the fundamental building block of the brain. Electrical signal that flows between these neurons provides the channel for communication between these neurons. A neuron in brain is connected to thousands of other neurons creating a neural network that form the basis for learning and memory (Gurney 1997). Figure 2 below shows different components of a neuron.

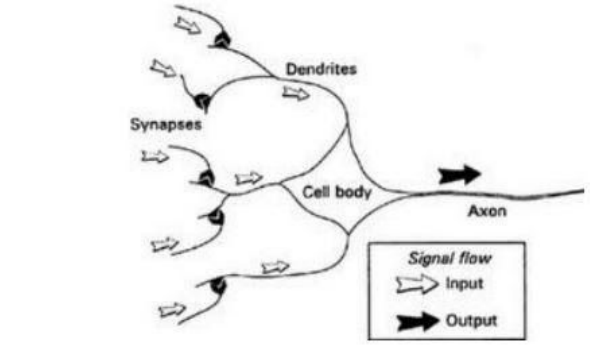

Figure 2 Different components of neuron (Gurney 1997)

As shown in the figure 2; biological neuron can be explained with input and output received and emitted by a single unit. When these neuron functions in unison they provide human with the possibility to learn and memorize. A resulting signal needs to pass certain threshold to be fired so that another neuron that are joined with the neuron gets impulse from previous neuron.

When hardware and software is organized to simulate how human brain learns and remembers patterns. It is called Neural Network. An artificial neural network; sometime referred as feed forward network due to each previous neuron being connected to next neuron is an interconnection of neurons (Maad, Esen and Alsaadi 2019).

2.1 Deep learning introduction and definitions

The history of AI dates to 1950 when the field of computer science was at nascent stage. The aim of early research in AI was to find out if it is possible to automate the intellectual tasks performed by humans. The initial approach to make a computer think was to handcraft large set of rules to manipulate knowledge to give the impression that computer is thinking. Such systems were popular until 1990 and known as Symbolic AI. This approach was only applicable when a problem is well defined such as playing chess, but it was not able to solve complex problem such as navigation for an agent in an unpredictable environment such as traffic (Chollet 2018).

Traditional programming approach is to add business rules by programmer to meet new business requirement but as the businesses are facing fierce competition hardcoding business logic can sometime be too late. Thus, businesses are prepared to take into use data generated in their system to gain new insight about product, services, and customer behavior. Data plays an important role in business strategy. Using appropriate machine learning models, it is possible to make high accuracy prediction which can help businesses to be ahead of competition. Failing to realize emerging and hidden opportunities and challenges that data can provide can be expensive for a business (Hurwitz and Daniel 2018).

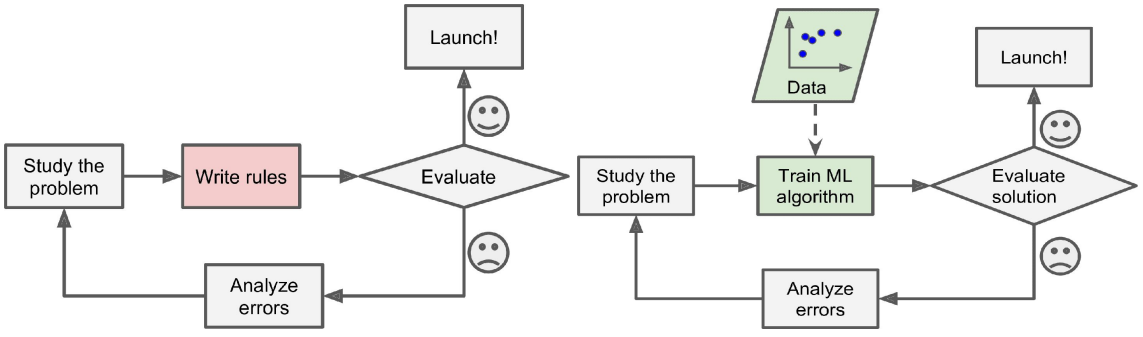

Figure 3 Traditional approach (left) vs Machine learning approach to solving problem(right) (Géron 2020)

Figure 3 above shows traditional approach which includes hardcoding logic vs the machine learning approach where algorithm is based on training on the data.

Any discussion on deep learning is incomplete without discussing machine learning. It is essential to understand what *learning* is. Machine learning includes three components: *input data*, *expected output* and a measurement of *how well an algorithm is doing*. This measure of *how well an algorithm is doing* is learning because it acts as feedback signal to steer the model towards expected output. Thus, a machine learning model is an engine that is fed with input data and which applies its learning to produce a meaningful output (Chollet 2018).

Machine learning has several definition depending upon the researcher and organization but core theme among all the definition is that machine learning is a science that consists of algorithm and processes to feed data gathered from various sources and infer new information based on the data (Faggella 2020). Machine learning works by running training on existing data so that possible associations are formed and once the training is completed then trained model is fed with actual data to gain insights. For example, for a weather predicting model, the training data could be the historical weather data gathered over time and the prediction could be made if it will rain on a particular day or not.

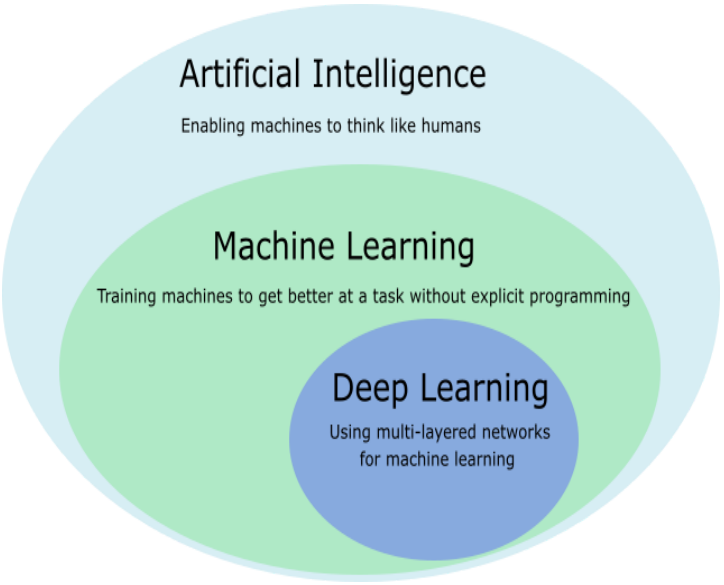

Figure 4 AI, Machine Learning and Deep Learning

As shown in Figure 4, Deep learning is a subset of machine learning which itself is subset of AI and has gained significant attention at present due increased coverage in media on its

application such as self-driving car, intelligent agent in department store and cashier free department store shopping. According to a research paper published by Microsoft, deep learning is defined as a machine learning technique that exploits several layers of non-linear information for both supervised and unsupervised feature extraction and transformation and for pattern reorganization and classification (Li and Dong 2014). For a long time it was difficult to devise model for feature extraction from raw data such as an image because it required domain knowledge in image processing to transform the image data into suitable representation, for example into pixels for an image, which would be fed to a classifier to for classification.

There are two types of learning namely supervised and unsupervised. Supervised learning is when training data includes both the input and expected result. For example, when training a model to recognize fruits, training data can include labelled image of fruits with expected result. Unsupervised learning is when expected result is not provided during training and thus can be used to cluster input data into classes. For example, if a model is fed with several hundreds of thousands of pictures of fruits then is able to cluster red colored oval item together i.e. apple (Donalek 2011).

Deep learning is a machine learning technique but takes different approach to learning. The *deep* in Deep learning refers to the successive layers of representation in neural network. So, a deep learning model is called neural network which are stacked upon one another as shown in Figure 5 below. Figure 5 has only single hidden layer but for any real-world working model there can be several hidden layers.

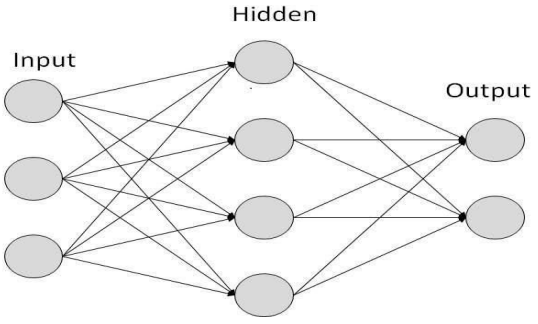

Figure 5 A neural network with a single hidden layer

Figure 5 above shows three fundamental components of a neural network namely: input layer, hidden layer, and output layer.

There has been massive media coverage about AI in the recent years and a common theme among those reports is that deep learning is modelled after biological brain. It is not entirely true because some concepts were implemented inspired by human brain but deep learning itself is not a representation of human brain (F. Chollet, Deep learning with Python 2018).

Deep learning has been widely used for Natural Language Processing (NLP), pattern recognition, image recognition, voice assistant, recommendation engine such as Facebook and Google (Wagner 2014). So, deep learning finds its application in domains with unstructured data as deep learning has proven to be able to find features in unstructured data as well. New domain of deep learning application has been identified in the realm of cyber security, product development, government workflow. Anomalies images detection is among the application of deep learning and there has been continued interest in this research area because finding anomalies can be a rewarding for a business that can result in cost and time saving.

Recent increase in deep learning is fueled by the availability of big data in all the fields of interest with the rise of Internet users in the world. Similarly, computing capability of machine was a major constraint in deep learning which has been solved by recent advancement in chip development which has made hardware cheaper and affordable. At present, it is possible for anyone to organize a deep learning experiment with few hundred dollars. More and more cloud services are geared towards providing the computing needed for running deep learning experiments (Khoshgoftaar, et al. 2015).

2.1.1 Input data

Input data lies at the heart of deep learning as it is the raw data that is used for training a model. At present there is abundance of data in different domain such as medical data, social media data, data collected by health services, social medias, government agencies, data from sensors and machine generated log data etc. With recent rise of Internet of

Things (IoT), there has been huge surge of data for a particular use case such as temperature and humidity data at home or data on amount of sunlight received at balcony. Such data can be used for both training and making predictions and improve the devices that controls the environment it resides on. Input for a neural network is any structured or unstructured data which can used for training in case of supervised learning and used for clustering to find representation in case of unsupervised learning.

Data for unsupervised learning can be utilized by deep learning algorithm to find patterns and representation that are not easily discoverable with traditional programming approach. So, instead of taking the traditional approach of writing rules with what to with data; deep learning is given available data and expected result. It is then for deep learning to formulate the rules. Traditional approach on data such as Symbolic AI where rules are handcrafted by a human or using decision tree and case-based reasoning fall short to utilize the potential of the data. For example, an image of dog is composed of various shape and features. So, deep learning algorithm uses such data to breakdown into simpler representation that can be used in other context such as object detection, classification etc. (Najafabadi, et al. 2015).

Even though there are several applications of deep learning; computers vision and image detection are amongst the most used application which has been possible due to abundance image data on the Internet. Even though there is abundance of image data available in the Internet, the challenge is to retrieve, index and organize the data that can be used to these algorithms. There are several open source datasets available to test and train deep learning models such as ImageNet, Open Images Dataset V6. ImageNet is amongst the most popular crowdsourced dataset which contains 14 million hand annotated image data. ImageNet organizes images based on semantic hierarchy. This dataset finds its importance when experimenting with new deep learning algorithm (Deng, et al. 2014).

Kaggle[6], a Google LLC subsidiary, is a popular community to publish dataset. The platform allows both organization and induvial to publish dataset. Not having to worry where to get data allows researchers focus on the problem.

[6] https://www.kaggle.com/datasets

2.1.2 Neuron and Backpropagation

A neuron is sometime referred as a perceptron in deep learning is placeholder for mathematical function. A neuron is the simplest processing unit for input information. A weight is multiplied to the input data and finally a bias value is added. Output from a neuron is represented by following formula.

$$\textbf{Output} = \sum \textbf{(input * weight)} + \textbf{bias}$$

The role of weight is to emphasis the important of an input. For example, output would be higher if weight is high thus signifying the importance of input. Similarly, bias is used to set a threshold to eliminate output that are not significant. The goal of training is to find appropriate values for these parameters.

Similarly, when an output is produced; it is necessary to understand how far the produced output is from expected output during training. A loss function calculates this score distance called loss score. Loss score plays an important role to adjust weights to improve the prediction. An optimizer uses the loss score to and calculates new weight for successive layer. A backpropagation algorithm is used by optimizer to calculate the new weight. (Chollet 2018). Figure 6 below shows the overview of Deep learning training.

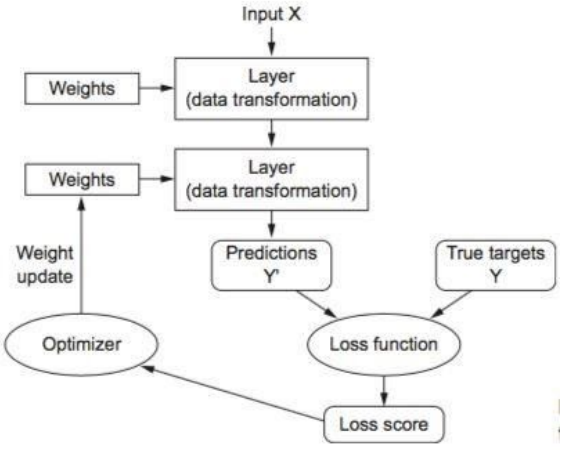

Figure 6 Deep learning overview (Source: Deep learning with Python)

Backpropagation is used to create relations between two neurons with series of failure and successes. Such relationship is stabilized during training which can be used for future task.

Generalization is reached when training is done. Likewise, these networks are also able to learn task like the ones that are already learnt. When a neuron activates itself in a certain way; it affects the subsequent neuron connected to it resulting in relations that have shape of fuzzy rules. If a neuron N_1 activates in X way (N_1 Activation value) then neuron N_2 connected to N_1 in feed forward network will be activated in X times Y where Y being the connection strength between these N_1 and N_2. (Buscema 1998)

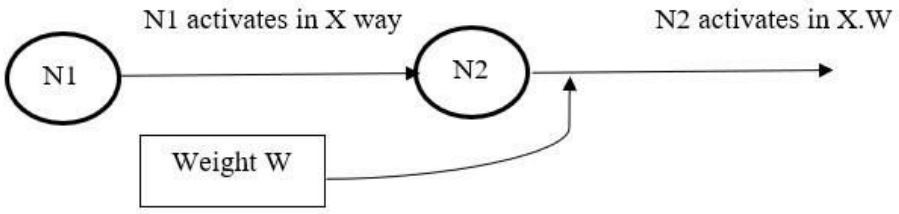

Figure 7 Relationship between two units in feed forward network

Figure 7 shows N1 when activated in certain way influences how N2 activates along with the weight. N1's activation is result of similar process as with the N2's activation thus BP is a stream of modifying Fuzzy Rules.

Backpropagation has been synonym to deep learning, but it is not a new concept. The first mention of backpropagation dates to 1960 but breakthrough work was done by Seppo Linnainmaa in this Master Thesis titled *The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors*. (Schmidhuber 2015).

2.1.3 Activation function

Weighted sum of input and biases can be anything between -infinity and +infinity. Thus, computation is necessary to determine if a neuron can be fired or no. Such computation is handled by an activation function as the name suggests. Some of the popular activation function are SoftMax, ReLu, Tahn etc. The process of learning depends upon use of correct activation function. ReLu is amongst the most used activation function.

According to universal function approximation theorem (UFAT), activation function is defined as a constant, bounded, and continuous function. Neural Networks with linear unit that are unbounded also conform to this definition. The performance of neural networks is influenced by optimization method used during training. The primary step towards building neural network is determining the type of activation function. This outlines the importance of activation functions on the performance of the network (Goyal, Goyal and Lall 2019).

Activation function can be both linear and non-linear based on the need to control output from neural network. Proper choice of activation is amongst the key factor to improve the results of neural network. Logistic functions were amongst the most common activation functions due to their thresholding behavior that resembles with how neuron is fired in the brain. ReLu (rectified linear unit) which is linear for positive input and zero for negative inputs and it was experimented that ReLu was able to improve results for image recognition problems with deep networks as it solved the problem of vanishing gradient. Despite solving the problem of vanishing gradient ReLu cut off the gradient when input is negative. Leaky ReLu was introduced to address this problem which has a small non-zero slope even for negative value which further improve image classification to surpass human-level performance on ImageNet task (Urban 2017).

Current researches in activation function is focused on adaptive activation functions (AAF) which can automatically select activation for each neuron from a predefined set. Such learning during training can be another milestone in the realm of activation function. Agostinelli et al. in their 2015 has experimentally demonstrated that a parametrized, piecewise linear activation function that can be learned independent by each neuron using gradient descent can help be useful to train deep neural networks to attain state-of-the-art performance for various deep learning tasks (Agostinelli, et al. 2015). Figure 8 shows commonly used activation functions for training deep neural networks.

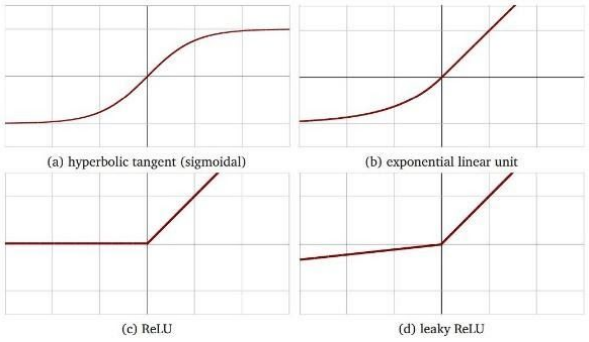

Figure 8 Activation functions most used in neural networks (Urban 2017)

2.1.4 Supervised and Unsupervised Learning

Supervised methods are used to discover relationship between input properties and output. For example, devising a model that can detect pictures of cat after training by feeding thousands of pictures of cats. The relationship is often referred to as model. Supervised learning finds application in classification, marketing, and finance. Classification and regression models are amongst the most popular supervised learning model. Regression models are used to make prediction based on available attributes such as the likelihood of summer being rainy or not based on historical data. Classification models are used to map input to predefined output. Training set is used to train model to recognize previously unseen examples. A test set is used to make testing on the model. The reliability is determined based on how well a model performs on the test data. Train data can include simple data such as numbers to complex data such as images, sound, videos etc. Based on the data at hand; training data is transformed into numerical representation using matrix (Maimon and Rokach 2020).

In unsupervised machine learning (UML), the goal is to detect pattern from the unlabelled data. Homogenous subgroup can be found from the unlabelled data based on the distance between the data. Dimensionality reduction is another application of unsupervised learning where the output from unsupervised learning can be used as input for supervised learning. Previously undetected pattern can be discovered in the absence of labelled data using UML.

2.1.5 Transfer learning

Humans can apply learning from one context into another context; thus, there is no need to learn each task from scratch. For example, learning to ride bicycle would help someone to learn to ride motorcycle and knowledge of using gears while riding motorcycle can be useful to drive a car as well as understanding of traffic rule learned while driving motorcycle can be applied to driving a car or truck. But deep learning algorithm works in isolation to solve a specific kind of problem and changing the feature-space also means training a different model from scratch. So, transfer learning is the research domain where learning gained from one domain can be used to solve problem in other similar domains. A pretrained model can be used to solve problem for another task (Sarkar 2018).

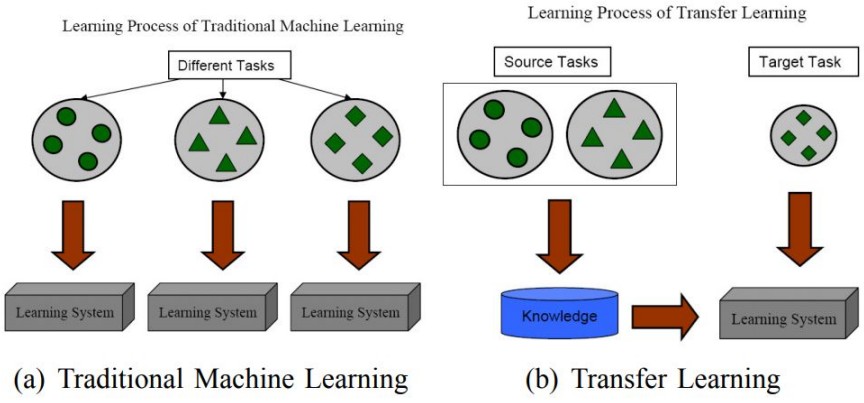

Figure 9 Traditional machine learning vs transfer learning (Pan and Yang 2010)

Figure 9 shows the difference between transfer learning in comparison to traditional machine learning approaches. Transfer learning allows find applications where previous training can be applied in the new related domain.

Transfer learning uses a pre-trained model using dataset like ImageNet[7] and main part of the weights of the model pre-trained on general purpose dataset remain frozen. Thus, keeping the general information domain independent. Only some last layers are retained on new domain specific dataset to extract new domain specific feature. Thus, binding domain

[7] http://www.image-net.org/

independent features of the pre-trained model with the domain specific classes. Such approach is relevant in case of classification task. This eliminates the need for running training from scratch from the new data. Features from the pre-trained model are useful for the task at hand. VGG[8] Model (developed by Oxford), Inception[9] Model (developed by Google) and ResNet[10](developed by Microsoft) are well-known model using Image data. (Puthige 2020).

Transfer learning enables knowledge transfer allowing faster training, brings possibility to use open source dataset such as ImageNet which consist of millions of labelled data.

2.2 Deep Neural Networks (DNN) Architectures

The concept of perceptron was put forward by Frank Rosenblatt. It can be defined as a machine that can perform simple operation and divide input into two parts – "yes" or "no", thus the result is always classification decision and it was modelled after the human neuron. (Lefkowitz 2019).

Figure 10 shows a perceptron which is a basic learning machine. Components of a perceptron includes input values (input data), weight and biases, weighted sum, and the resulting activation function.

When multiple perceptron is stacked together to solve complex problem like object detection, image classification etc. then it is called Multilayer perceptron (MLP). DNN and MLP are interchangeably used when referring to a neural network. However, MLP are always feedforward while DNN can have loop (Perceptrons and Multi-Layer Perceptrons: The Artificial Neuron at the Core of Deep Learning 2020).

DNN are thus any neural networks that consist of Input layer (Input data), hidden layer (stacked neurons) and output layer that produces output. For example, Convolutional Neural Network (CNN) is a DNN that is widely for image processing, Recurrent Neural

[8] http://www.robots.ox.ac.uk/~vgg/research/very_deep/
[9] https://cloud.google.com/tpu/docs/inception-v3-advanced
[10] https://docs.microsoft.com/en-us/azure/machine-learning/algorithm-module-reference/resnet

Network(RNN) is widely used for machine translation, NLP and are good at processing sequential data (Moolayil 2019). Figure 11 shows components of DNN.

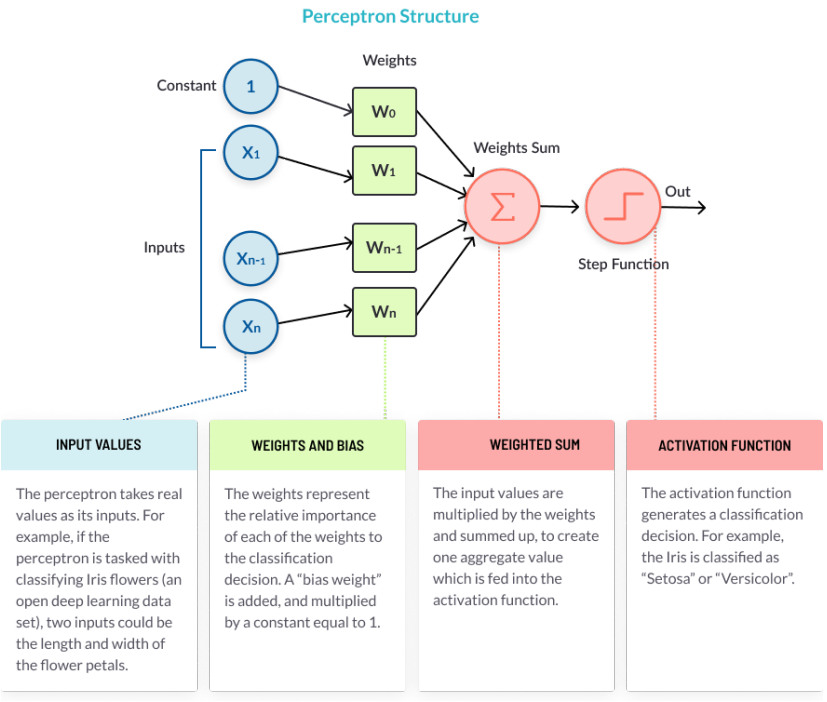

Figure 10 Structure of perceptron (Perceptrons and Multi-Layer Perceptrons: The Artificial Neuron at the Core of Deep Learning 2020)

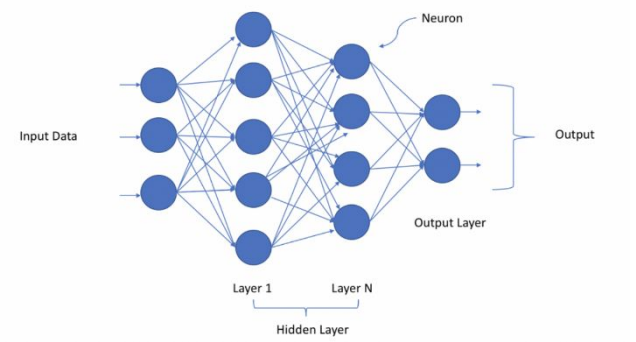

Figure 11 Structure of DNN (Moolayil 2019)

2.2.1 Convolutional Neural Networks

A Convolutional Neural Network (CNN) is a Deep Learning algorithm that differentiates images based on the assigned importance through learnable weights and biases to various aspects/objects in the image. CNN which are the extension of neural networks for treating images; is shown to be effective in many branches of vision applications, such as object recognition, segmentation, and classification. Spatially shared weights and spatial pooling are the two main features that make them extremely useful for image applications. As shown in Figure 12, CNN focus on low level features and edges is what makes CNN efficient in image classification when there are multiple objects in an image. (Gu, et al. 2017)

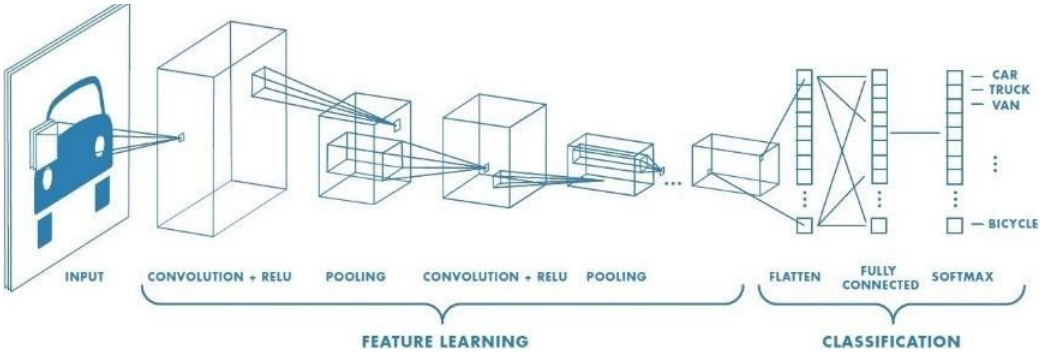

Figure 12 Feature learning and classification in CNN (Deshpande 2019)

The convolutional neural networks (CNNs) play a dominant role for processing visual-related problems. CNNs are biologically inspired and multilayer classes of deep learning models that use a single neural network trained end to end from raw image pixel values to classifier outputs. Various studies have shown that CNNs can effectively solve problems where other methods have failed or had limited success. CNNs use and success are studied and verified in many vision related tasks, including image classification, object detection, scene labeling, house number digit classification and face recognition. CNNs has been forwarded as an effective method for understanding visual image content, giving some state-of-the-art results on visual image classification and other visual-related problems (Hu, et al. 2014).

The inspiration for CNN comes from biology where some neuron in visual cortex are fired when certain feature is present in visual field. It is based on the experiment performed by Wiesel and Hubel in 1962. (Deshpande 2019)

2.2.2 Recurrent Neural Networks (RNN)

In recurrent neural networks the decision is influenced by the things learnt in the past, neural network remember the things that were fed during the training thus making an impact in the final output. Unlike other neural network, an RNN can process the sequential inputs by having a recurrent hidden state whose activation at each step depends on that of the previous step. Although RNN is an important branch of deep learning family and has promising applications, one of the common issues in conducting RNNs research is the difficulty to train the RNNs to deal with long-term sequential data, as the gradients tend to vanish (Mou, Ghamisi and Xiang 2017).

Since RNNs have a "memory" function, which can remember the information of the past state and use the information in the future learning. RNNs in the beginning were only used for single dimensional (1D) sequence learning such as speech and language recognition. Since image data is of higher (two-dimensional or three-dimensional), and the conventional RNNs are not suitable for image context learning. In order to get the suitable sequence from the image, some methods such as multi-dimensional recurrent neural network (MDRNNs) were proposed for high-dimensional data learning (Cheng and Pun 2018).

2.2.3 Recursive Neural Networks

Recursive Neural Networks is commonly found in the inputs of different modalities such as natural scene images or natural language sentences. RNNs not only help to understand the recursive structure of the image but also show how the images and sentences interact to form a whole. (E. Francesconi 1997)

One major attraction for using recursive neural networks (RNNs) in neural network study is that, RNNs are able to process structured inputs by repeatedly applying the same neural network at each node of a directed acyclic graph (DAG) (Socher, Manning and Ng n.d.).

Study conducted by (Bianchini, et al. 2005) has shown that by using graph-based representation of images that combines structural and symbolic visual features and then processing by RNNs, in order to establish the possible presence and the position of faces inside the image can be an effective way to localize the faces. This method can be very useful in surveillance purpose in meeting places, like airports, railway stations or to restrict the admission to reserved areas.

2.2.4 Autoencoders

Autoencoders are defined as learning circuit for data transformation with minimal amount of distortion. Autoencoders belongs to Generative models and play an important role in unsupervised learning since Hiton and PDP group devised autoencoder in 1980s for building system that can learn from the data. Autoencoder compress the data, encode data, and learns how to construct the data again from the encoded representation which closely resembles the original representation (Baldi 2012). Thus, fundamentally autoencoders consist of 2 components, an encoder and a decoder as shown in Figure 13 that shows autoencoder for MNIST data.

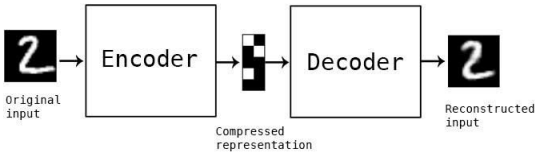

Figure 13 Autoencoder for MNIST data (F. Chollet 2016)

With autoencoders data is learned from example instead of being engineered by human and compression and decompression can be done only on the data that is previously seen before. For example, autoencoders trained for recognizing the breed of cat would perform poorly to on cat videos. Loss is always associated with autoencoders which means the output is degraded compared to original data. Autoencoder can be used for anomaly

detection among other application because encoding is based on similar feature and loss on reconstruction is low for similar image while high in outlier.

Autoencoders also can be used in the cybersecurity domain and have been focus of research to find semi-supervised solution based on autoencoders that can use both labelled data as well as learn patterns from new cybersecurity threat. Anomaly detection has been a major challenge in cyber security due sophisticated attacks. Yousefi-Azar et. Al 2018 have presented novel method to use autoencoder to classify and detect malware. Technique using autoencoder to provide discriminative feature that have stood out compared to other engineering approaches (Yousoufi-Azar, et al. 2018).

Figure 14 shows the architecture of an autoencoder, when h is the hidden layer and the encoder part is to be represented by function $h = f(x)$, then the decoder part that performs the reconstruction can be represented by function $r=g(h)$. Traditionally autoencoders were used for feature learning and/or dimensionality reduction. Autoencoders are used to reconstruct its input and finds applications in unsupervised learning such as clustering. (Goodfellow, Bengio and Courville, Autoencoders 2016).

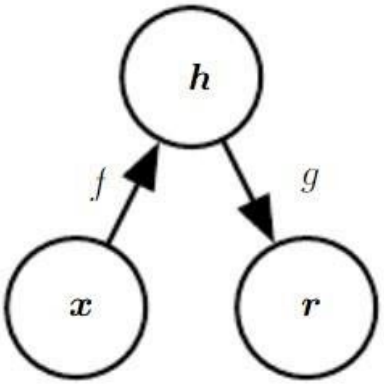

Figure 14 General structure of an autoencoder which consist of two components the encoder f (mapping x to h) and decoder g (mapping h to r) (Goodfellow, Bengio and Courville, Autoencoders 2016).

Different types of autoencoders are discussed below:

1. Denoising Autoencoder

   Used for adding noise to the input to avoid copying of input to output without learning data feature. Partially corrupted input is used for training to obtain undistorted original input. Figure 15 shows Denoising Autoencoder.

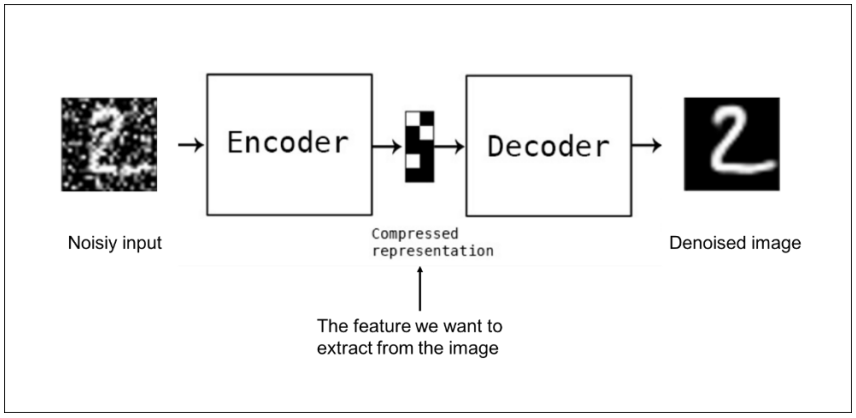

Figure 15 Denoising autoencoder

2. Sparse Autoencoder

Used for feature learning in unlabelled data and contains a single hidden layer and. Highest activation is taken while the rest are zero out in the hidden layer forcing use of reduced number of hidden layers. Figure 16 shows Sparse Autoencoder.

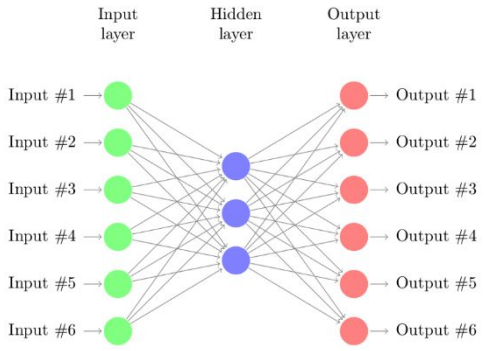

Figure 16 Sparse autoencoder

3. Deep Autoencoder

Fully connected DNN with a bottleneck layer and used for transforming high dimensional input lower dimensional input as bottleneck feature and eventually transform the bottleneck feature into high dimensional representation. Figure 17 shows a Deep Autoencoder.

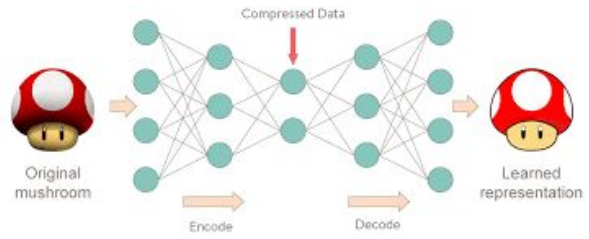

Figure 17 Deep autoencoder

4. Contractive Autoencoder

   Contractive autoencoder are used as a regularization technique like denoising and sparse autoencoders. These are less prone to small variation in the data.

5. Undercomplete Autoencoder

   Undercomplete autoencoders are used to capture the principal feature from input data. The autoencoder have smaller dimension for hidden layer that helps to minimize loss function. Figure 18 shows a Deep Autoencoder.

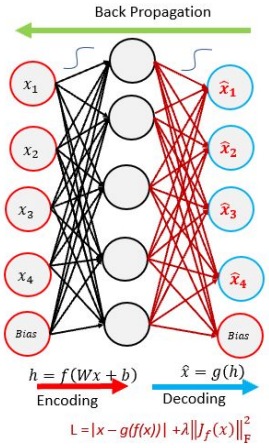

Figure 18 Undercomplete Autoencoder

6. Convolutional Autoencoder

Convolutional autoencoder encodes input in a simple signal and reconstruction is attempted from the input. This autoencoder is the state-of-art technique for unsupervised learning. They can be used for higher image size once trained. Figure 19 shows a Convolutional Autoencoder.

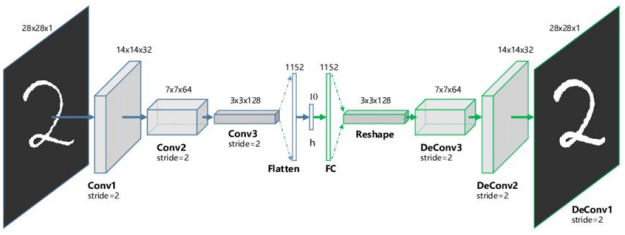

Figure 19 Convolutional Autoencoder

2.2.5 Generative Adversarial Networks (GAN)

Ian Goodfellow, researcher at Google Brain[11], along with his colleagues invented GAN which amongst the revolutionary inventions in the field of AI. GAN is a deep learning framework where two networks are simultaneously trained; one model is Generative

[11] https://research.google/teams/brain/

model, and another is Discriminative model. The training is performed such that Generative model G is assisted by Discriminative model to train based on real data. Both networks are trained on the same dataset. For example, if a Generative model G also known as *generator* generate new samples and second model D also known as *discriminator* compares the generated samples with the original sample and makes prediction if the produced sample is real or not. This process continues until G can create samples such that *discriminator* no longer can tell generated samples from real one (Giles 2018). As shown in figure 20, *discriminator* acts guide to help *generator* reach a goal of making almost identical samples by providing feedback on the generated samples.

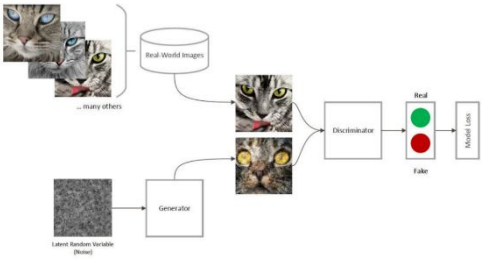

Figure 20 Architecture of GAN (Contrascere and Schwalm 2019)

3 Anomaly Detection

3.1 Anomalies in data

Anomalies or outlier in data refers to data points that differs significantly from normal groups. Anomalies can occur due to incorrect data collection as well as exceptional situation when such data point occurs hinting to corner cases that might reveal new characteristics of the data. Anomalies detection are techniques applied to find such pattern that do not confirm to normal behavior. Due to rare occurrence of anomalies in data; anomalies often give the impression as if they originated from different mechanism. When the observations are one dimensional it is easy to identify anomalies but with complex data such as image and video; anomalies detection can get challenging (Rosebrock 2020).

For example, detecting anomalies in rainfall within a year in a place can be easier when compared to anomalies detection of medical images. Figure 21 shows a single data point that is at a significant distance from normal data point signifying an exceptional event.

Figure 21 Anomaly data

Anomalies detection finds applications in various domain such as finance, cyber-security, fault detection and surveillance. Anomaly detection is important because anomalous data is often critical and overlooked. For example; anomalous network traffic to and from a

machine could be due to the machine being compromised and sending out sensitive to an unauthorized party (Chandola, Banerjee and Kumar 2009).

Anomaly detection differs from noise detection which is removal of unwanted data to help understand the data at hand better. Previously unobserved data patterns during machine learning training identified to discover new pattern within a given dataset which is known as novelty detection. Such patterns are eventually incorporated into the normal model after detection (Markou and Singh 2003).

Image anomalies find application in industrial setting to detect defected product line, insurance companies can use the image-based anomaly detection to find if a claim for a crack in glass is anomalous or not i.e. if it is intentional or not. If a claim is made for a broken glass and based on the point of impact it can be detected if it is normal to have the type of cracks. Video based anomaly are used by government agencies to identify unusual behavior in the public.

Chakravarty, et al. 2020 performed an experiment on patrolling robot with the ability to detect anomaly using image anomaly detection. A patrolling robot was designed which is autonomous and mobile that can repeat certain routes and detect any visual anomalies. A panaromic visual sensor helped the robot navigate. Experiment showed that robot was able to detect obstacles on it's route using image anomlay detection.

Anomalies can be classified into three categories broadly as discussed below (Baddar, Merio and Migliardi 2014):

1. Point anomalies

   Single occurrence of an event that does not fit the ordinary. Credit card fraud often fall in this anomaly category. When a credit card is compromised, often the perpetrator goes into using the stolen card into different places as soon as possible and such behavior can be registered as an anomaly. Figure 22 shows single point anomaly which differ from rest of the data point significantly.

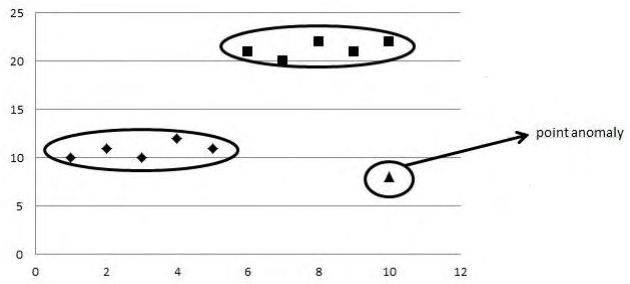

Figure 22 Example of single point anomaly (Baddar, Merio and Migliardi 2014)

2. Contextual anomalies

   Context matters when detecting anomaly. Anomalies that are context sensitive are contextual anomalies. If a person from Finland suddenly starts using credit card in Ethiopia, then it could trigger anomaly based on the person's connection with Ethiopia and the person's travel history. Such behavior depends upon context of if the person has any connection to Ethiopia. Figure 23 shows contextual anomaly which is context sensitive, data point can differ based on various factor such as time and location.

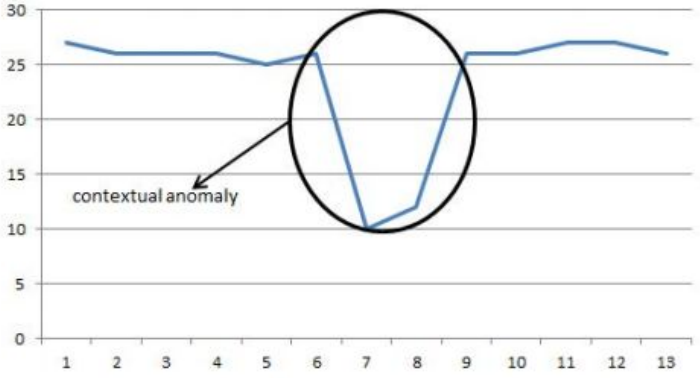

Figure 23 Example of contextual anomaly (Baddar, Merio and Migliardi 2014)

3. Collective anomalies

   Collection of data point collectively contribute to anomalies for example during a Distributed Denial of Service attack, several IP addresses are requesting a resource

taking the server down. Figure 24 shows example of collective anomaly which are based on data point from several sources that suggest anomalies.

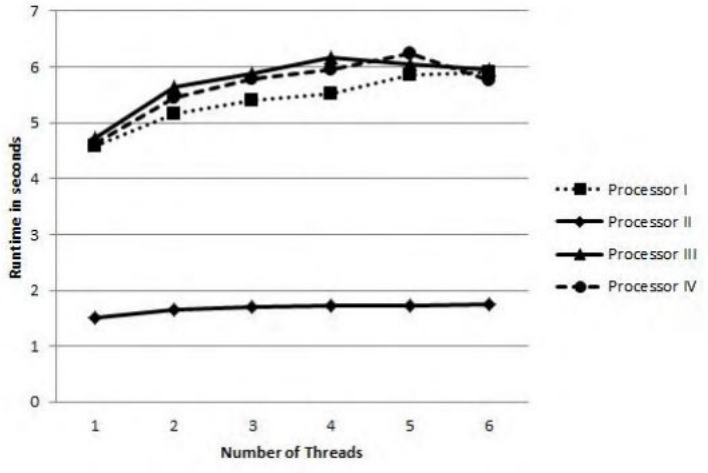

Figure 24 Example of collective anomaly illustrated by the values of Processor II (Baddar, Merio and Migliardi 2014)

3.2 Department store

Images as well as video anomaly detection in department store can help department store understand product placement better, finding misplaced products easily and deal with possible theft and damaging of items. Similarly, application can include detection of dangerous situation such as unusual stacking of items, customer trying to reach forbidden areas etc.

Companies like SES-imagotag which traditionally provided retail with shelf labeling solution provide solution to detect anomalies in shelf by detecting misplaced product. The solution is based on product tag that is read by camera placed on top of shelf (Automatic Realogram & Real Time Stock-out Detection 2018). Finding misplaced item in shelf is not enough by merely object detection due to additional context needed to determine the correct location of the misplaced item. For example, if two different kind of bowls are stacked after another then it can be a challenge to find the misplaced item. Cai, et al. 2020

suggested that bounding box to detect item is not sufficient in retail store context where items are stacked upon one another. In such retail store context they purpose a new method to count object using deep learning namely Localization and counting shortly *LoCount* which is useful to count objects. But *LoCount* could be also used to detect anomalies in cases where an item is misplaced. For example; if a bowl of different sizes are mixed together by a customer in a department store it would be possible to count the misplaced item and find the correct place where it should be found.

AI is intergated into department store to assist in Sales and CRM applications, customer recommentations, payment and payment services etc. Each of these fields can find use of image anomaly detection. For example, in logistics image based anomalies detection can help if a packet is broken or leaky. Store management can utilize image based anomalies detection can help find misplaced good and find the correct location of a good. Similary, wrong price is applied to an item my customer image based anomalies detection can alert the cashier. If a customer applies price sticker for potatoes for avocado then it is a loss for a store when avocado is expensive than potatoes and customer is in loss when potatoes is expensive than avocado. Such detection can bring integrity and trust to customer and the store.

Finnish company Solteq Oyj[12] developed robot that has the capacity of navigating a department store to help recognize products, empty or hole shelves and incorrect price location. Such autonomous robots have open path to integrating anomaly detection system into them (Solteq Retail Robot 2020).

3.3 Disease and injury detection

Anomaly detection finds various application in disease detection e.g. cancer detection, micro-calcification which are a small deposit of calcium that usually are implies the presence of cancer cells in breast, anomalies in organs such heart, lung etc. Similarly, in the context of injury detection and misdiagnosis anomaly detection can provide doctors and therapist involved with important information such as region to operate, recovery

[12] https://www.solteq.com/

approaches to take and amount of anesthesia to use during an operation. It is estimated that 12 million Americans are yearly according to the study conducted in 2014. Despite ample data, it was noticed that doctors missed crucial details making treatment more expensive both financially and psychologically. Such misdiagnosis consisted of serious cases such as colon cancer and lung cancer. Such misdiagnosis can pose risk to patient health and wellbeing. (Singh, Meyer and Thomas 2014).

Automatic anomaly detection of tumors has been a subject of long research which dates to 1973 when foundational work was done by Sklansky and Ballard to detect anomalies in radiographs. The interest in the field has been risen at present due to availability of historical data and advancement in computing and deep learning technology. AD in medical field is done by one of the three methods: classification with two classes normal and anomaly, segmentation technique which includes extracting feature from certain region in an image and content based approach where anomalous samples stored in database are compared with local samples and distance is calculated to determine anomalies if any (Crispi, et al. 2009).

AD services specialized for medical use can be devised using both supervised learning and unsupervised learning. There is abundant of historical that can be used to discover new knowledge using clustering. Similarly, same data can also use to modelling normality and anomalies in supervised learning.

3.4 Road condition detection

Good road condition is vital to the economy and society of a nation as inhabitants rely heavily on road for movement. So, maintaining a good road quality is beneficial for public and government. Traditional road quality monitoring includes human inspection and reporting obtained from public. Bad road condition can result in mistrust for government and in some cases increase overhead due to lawsuit. In Taiwan, state paid 130 million NTD (4.4 million USD) between 2005 and 2007 (Tai, Chan and Hsu 2010)

There are programs to monitor road condition that focuses to analyses the road condition in different countries such as Long-Term Pavement Performance that collects, store and

perform analysis on the road condition in the United State and Canadian road networks. Google has an imitative in Poland to discover potholes using mobile phone accelerometer that detects shocks to determine potholes. There are three most common approaches to detect anomalies in roach namely: 3D-reconstruction based, vibration based and 2D- image based using computer vision. 3D-reconstuction based method used a laser.

3.5 Anomalies detection techniques

Anomalies being data points that do not conform to normal behavior; the first approach that can be taken to detect such data points is to define normal data point groups. Thus, anything that does not belong to normal group can be classified as abnormal. Such approach is flawed due following challenges:

1. Defining region that take into consideration every possible normal deportment is challenging and difficult. There is no clear boundary between normal and anomalous pattern. Putting a hard boundary could make normal data points that are close boundary as anomalous and vice-versa.
2. Malicious actions such as sniffing traffic in a network usually adapt themselves to disguise into the normal behavior; thus, detecting and categorizing such anomalies is not possible with simple approach of defining border between normal and anomalous data.
3. There is scarcity of labelled data that is required for validation and training
4. Anomalous data can disguise as noise and discarded which makes it hard to discover (Chandola, Banerjee and Kumar 2009)

Presence of above difficulties in anomalies detection makes anomalies detection a difficult problem. Different factors such as nature of data, presence of availability of labelled data and the type of anomalies to be detected plays important role to determine correct anomalies detection technique. Each domain requires its own anomaly detection technique, so it is difficult to make one generic solution. Figure 25 shows different aspects associated with any detection technique.

In the context of department store, it is possible to classify each product which would make item detection easy and straightforward. However, a department store contains large number of products and training of object detection becomes a non-trivial task. Thus, the possible solution to the problem is to detect anomaly in a product area on a shelf without actual detection of items and their classification. Challenging cases of anomalies in dynamic context such as a department store are not based on classification due to limitation on data annotation due to presence of large number of items to classify, particular occurrence of a particular anomaly is rare , thus capturing samples is difficult. It is also possible that anomaly has never occurred before. Such limitation makes classification cumbersome and unsupervised learning-based approach is appropriate.

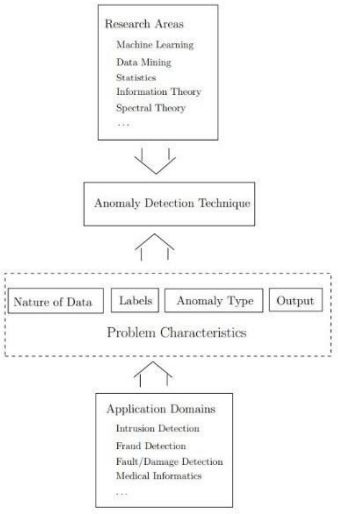

Figure 25 Components associated with anomaly detection (Chandola, Banerjee and Kumar 2009)

3.6 Image anomalies detection using deep learning

Visual anomaly detection provides assessment for visual defect in a visual entity. The visual entity could be an image or video. For example, finding puncture in a vehicle tire, finding the anomalies in the X-ray of chest to detect tuberculosis etc. include image anomaly detection. According to Minhas et. al 2019, availability of labelled anomaly training data and low availability of anomaly instances are amongst the major challenges to

study supervised anomaly detection. Both supervised and unsupervised methods can be applied to images anomaly detection however truly unsupervised anomaly detection techniques are rare to non-existent. Generative model i.e. Autoencoders (AE), Generative Adversarial Networks (GAN) can be used to identify normal data then based on the deviation from the normal distribution, anomalous occurrences can be calculated. Supervised approach includes using labeled training data using CNN where two classes are used for anomalous data and normal data. Due to need for high volume of data; it is often difficult to train such model from scratch. So, appropriate method would be to use Transfer learning (Minhas and Zelek 2019).

Image-based anomaly detection using deep learning can be approached in three ways according to (Pang, et al. 2020) as shown in Figure 26 below:

a. Deep Learning for Feature Extraction

   It is the primary application of deep learning for anomaly detection which works by leveraging the power of deep learning to extract low-dimensional feature representations. Feature extraction is disjoint from anomaly scoring.

b. Learning Feature Representation of Normality

   Learn normal situation thus paving path for anomalous situation. This includes the use of autoencoders which involves data compression that helps to learn important features and eventually using the loss function anomaly can be detected.

c. End-to-end Anomaly Score learning

   The approach uses neural network to measure anomaly score using the novel loss function that drive the anomaly scoring network

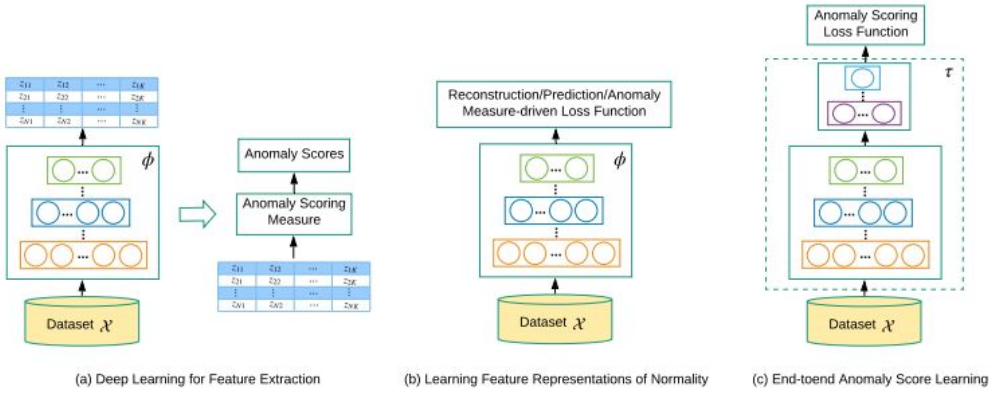

Figure 26 Conceptual Frameworks of Three Main Deep Anomaly Detection Approaches (Pang, et al. 2020)

3.7 Kernel Density Estimation

Kernel Density Estimation (KDE) is a statistical tool used for estimation of probability density function which enables the user to better analyse probability distribution compared to traditional histogram. KDE technique produces smooth estimate of the probability density-function and uses all the sample points (Węglarczyk 2018).

Image dataset can be rendered as collection of vectors, KDE can measure the density of each vector space occupied by the training data. When new anomaly observation is encounter, density estimation can be made to measure the distance of the anomaly dataset and training dataset. It is however not possible to apply this method to unprocessed images because the dimensionality would be too high to which makes distinctions between distances severely difficult. For example, an image with dimension 100x100 pixels implies a vector space of thirty thousand dimensions. It is not possible to measure distance or density among such large dimensions due to the '*curse of dimensionality*' it is not possible to measure measuring distance or density in such space (Wells 2020).

4 Image Anomalies Detection experiment

Conducting a deep learning experiment requires domain knowledge and identification of problem. This thesis explores applications of images anomalies detection in department store and presents with other possible areas where images anomaly detection could be used namely disease diagnosis, security, and road condition monitoring.

The aim of any deep learning experiment is to obtain minimum generalization error so that a model can produce good generalization result from training set. A model should not memorize the training data instead learn underlying rule in the data. So, that a rule is obtained so that unseen data can be generalized as well. So, the goal is to find the right hyperparameters that can optimize the networks.

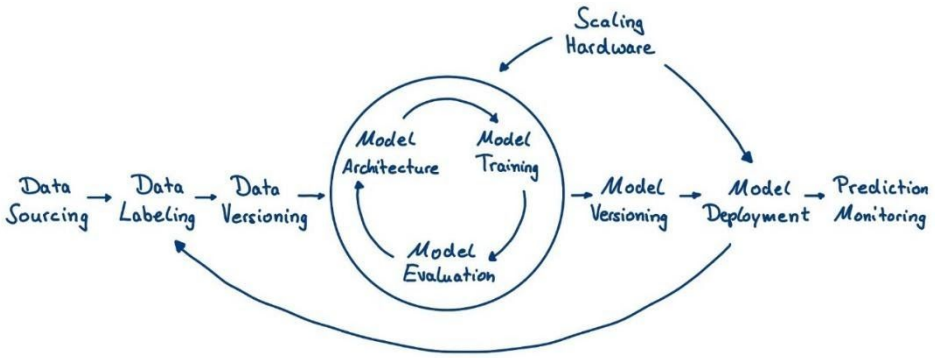

Figure 27 Deep learning experiment lifecycle (Ruban 2018)

Figure 27 shows the lifecycle of deep learning experiment that involves community accepted processes that are followed to conduct a deep learning experiment. It provides with an iterative process from data collection to prediction monitoring. Combination of hardware and software are required to facilitate such lifecycle when conducting large scale applications. But for this experiment we will perform steps such as data sourcing, data labelling, data versioning, prediction monitoring manually.

Following examples demonstrate the different anomalous situation that can be observed at a department store.

Figure 28 Example #1 anomalous product placement in department store

Figure 28 shows a simple anomalous product placement scenario where a chocolate pack is left in dog food section. Such anomalous situation is manually handled by human in department store.

Figure 29 Example #2 anomalous product placement in department store

Figure 29 shows a rather complex situation where washing liquid with different properties are mixed. These items share some properties, but anomalous item has its own placement.

Figure 30 Example #3 anomalous product placement in department store

Figure 30 shows a more complex form of anomaly where the item is same colour and both the items share common attributes in terms of packaging, but they differ in content.

4.1 Dataset

Dataset for the experiment was gathered by capturing product images from different department store in Helsinki using a smartphone. Photos were taken with the smartphone camera and stored in Google Drive. Training dataset consisted 50 different images product areas taken in different orientation and 10 images were used for testing which contained 5 anomalous images and 5 normal images. Photos were taken from different angles, in combination with neighboring products in shelf and product itself. Training was done for 3 different products namely apple, washing liquid and cold drink (Coca cola 2-liter jar). Anomalous pictures for checking the prediction were taken by placing an item that does not belong to shelf. For example, placing a chocolate bar in a shelf that is intended for washing liquid, mixing bananas with apple, and placing cold drinks of two different brands together.

Figure 31 Overview of dataset organization

Figure 31 shows how the dataset is organized in Google Drive[13] which is used as storage for the dataset for running the prototype experiment.

[13] https://www.google.com/intl/en/drive/

Figure 32 Normal data vs. anomaly data for Coca-Cola in a store shelve. Top image shows coca cola plastic jars without any misplaced item and it is the expected condition for the shelve. The bottom image shows addition of bottle from another company with completely different colour of the liquid representing anomaly. Similarly, a more complex example for anomaly is the addition of drink with same colour but different labelling.

4.2 Experiment design

Once data collection was completed; the platform to perform the experiment was decided. Google Colab[14] is cloud platform that provides Jupyter notebook service without the need to setup anything. Experiment script can be written in Python using the browser. Computing resources are available with possibility to purchase additional computing

[14] https://colab.research.google.com/

resources if necessary. Google drive was mounted to read dataset as well as write logs and models from training. Simple Python command line interface application was developed to check anomalies in image stored in google drive.

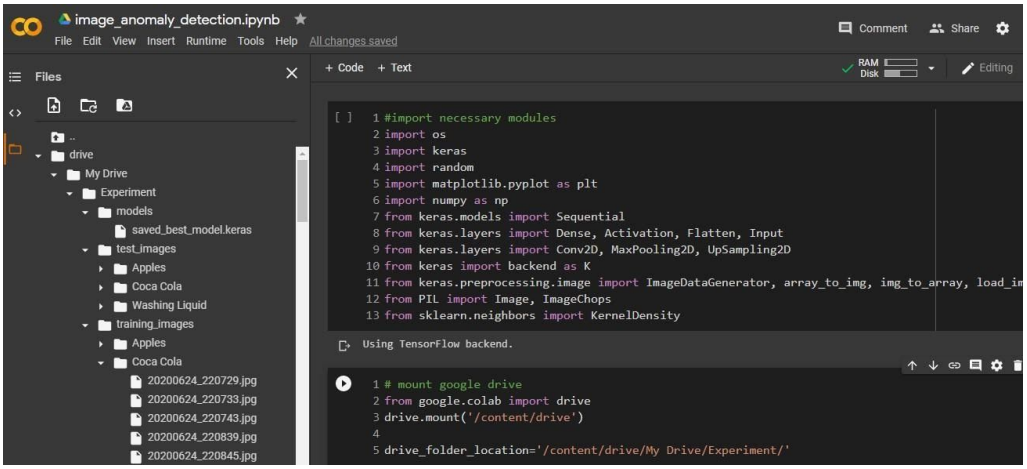

Figure 33 Google Colab for running experiment script

Figure 33 above shows Google Colab interface used for running experiment. Sequential API from Keras is used for the experiment to build CNN. Sequential API allows to add convolutional layer one after another in a stack. Sequential API has some limitations such as it is not possible to define multiple input sources and output destinations, but such limitation does not apply to this experiment. ImageGenerator is used to load data from Google Drive folder and transform each image into a fixed size from different sizes uploaded while collecting data. This gives the flexibility to provide any size of image to the dataset folder. ImageGenerator allows random transformations and normalization during training which is useful in scenario where the number of datasets is small. ImageGenerator is used to generate batches of augmented data which replaces original batch with new batch which are randomly transformed. (F. Chollet, Building powerful image classification models using very little data 2016). Conv2D Keras convolutional layer is stacked along with max pooling for dimensionality reduction. Mean Square Error (MSE) is used as metrics for evaluating the performance of the network. The model that performed best on the validation data is stored for making future predictions. Finally, predictions are made on new data using the saved model.

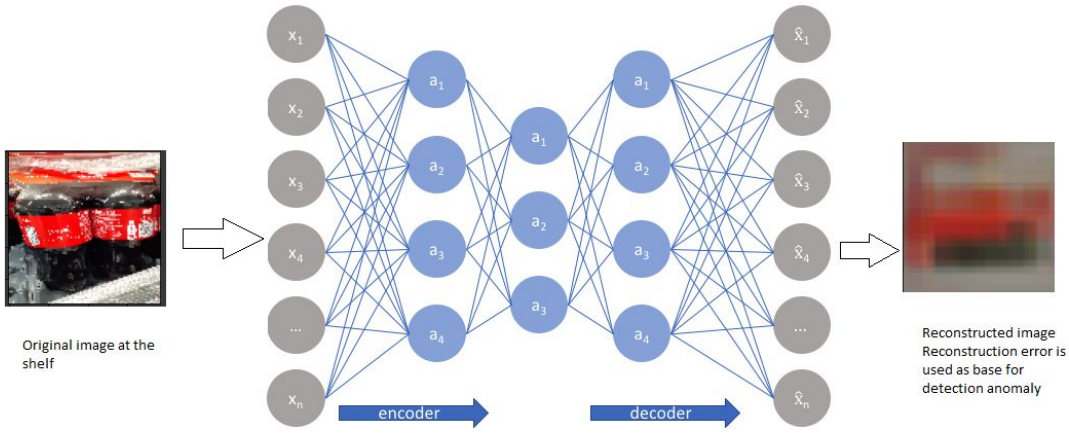

Figure 34 Autoencoder architecture used for the prototype experiment

Figure 34 shows the basic architecture used for the experiment. Deep convolutional autoencoder was used for the experiment with 5 conv2D layer. The threshold for accurate prediction was set so that if MSE between the validation images and anomalous images is more than 0.5 then it is considered anomalous.

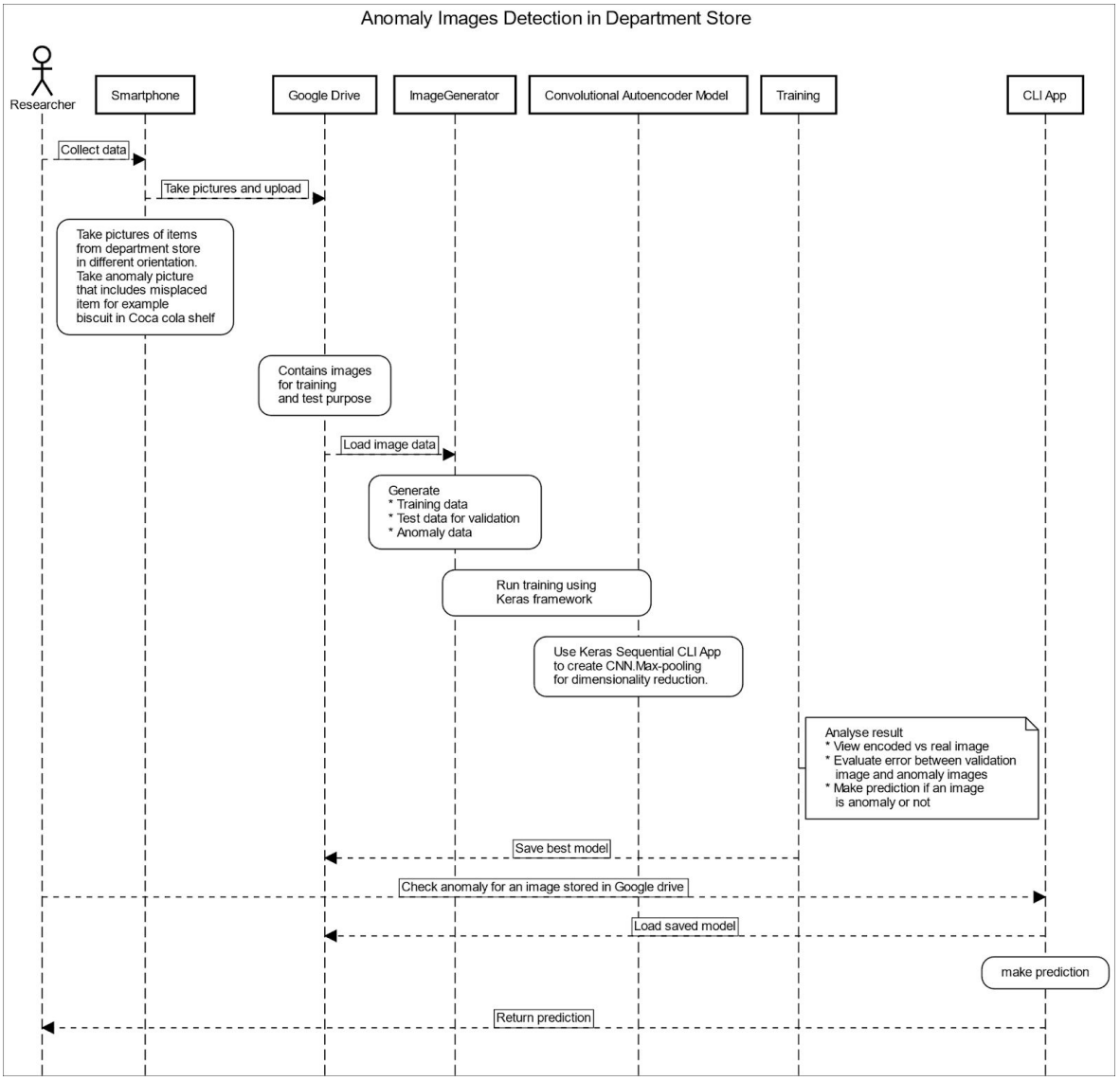

Figure 35 Experiment design

Figure 35 shows the experiment design sequence diagram showing different steps involved when running the experiment script. Once training is completed, predictions on the new data is made by providing path to the new image file to the command line script. Anomaly is detected by using autoencoder along with Kernel Density Estimation.

Table 2 Experiment parameters

Parameter name	**Value**	**Description**

Activation	Relu	Activation function used on until the last layer
Activation	Sigmoid	Activation function used in the last layer to determine if an image is anomaly or not as the results fall between 0 and 1
Optimizer	Adadelta	Used for adapting learning rates based on a moving window of gradient updates, instead of accumulating all past gradients
Loss function	Mean Square Error (MSE)	Calculate the loss during training

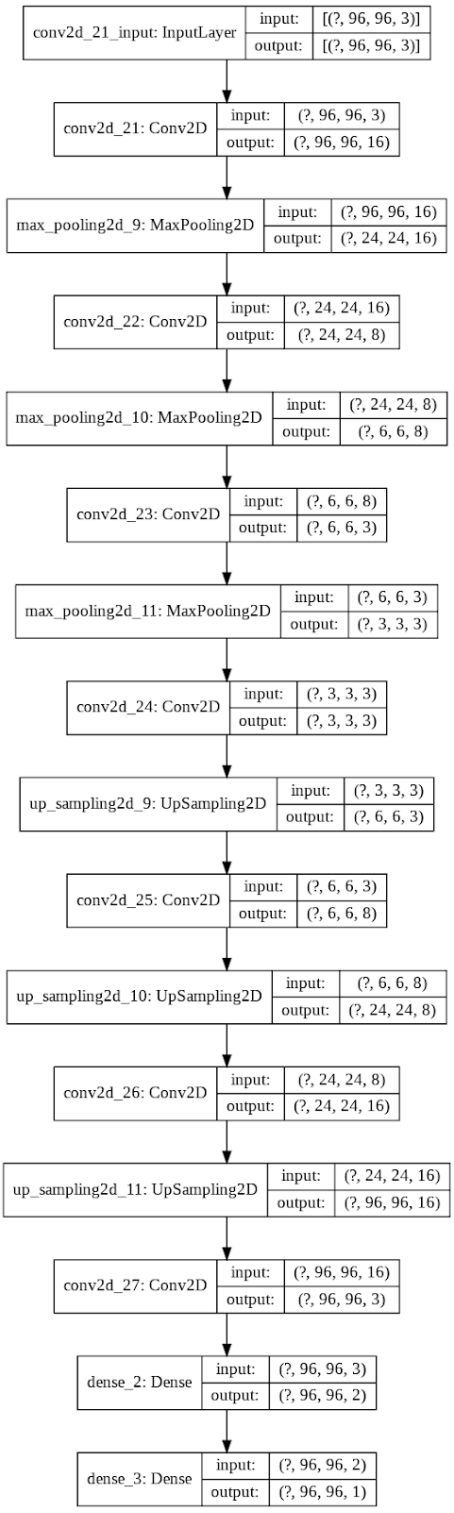

Figure 36 Model visualization obtained from Keras

5 Result

This section illustrates the result obtained from the experiment conducted for this thesis. The results demonstrate training results and anomaly detection results obtained for detecting anomalies in shelves which contains Coca-Cola. When training was done with 20 epochs then it clearly shows overfitting and accuracy does not increase after certain epochs. Similarly training loss and validation loss decrease with increasing epochs. A slight increase in loss in the validation data compared to the training data signifies overfitting.

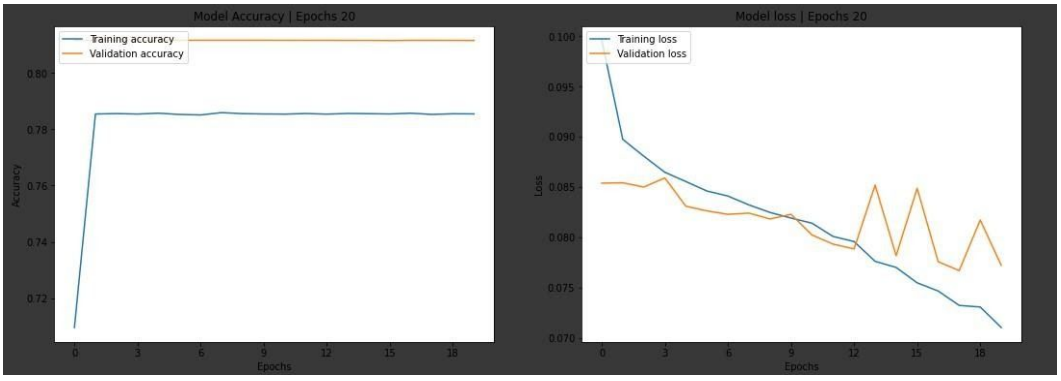

Figure 37 Accuracy and loss during training and validation

Similarly, the image generated by the decoder is as below in Figure 38. The decoded image is blur which simplifies that there is still need for further training.

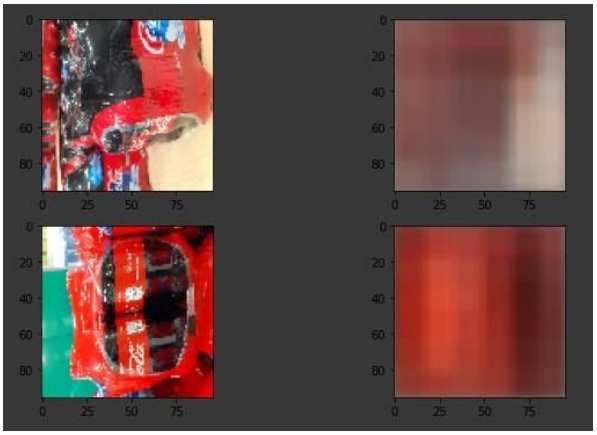

Figure 38 Example of actual vs decoded image for coca cola after 20 epochs

When anomalous images were predicted using the model then the accuracy increased compared to previous iteration but do not provide high level of confidence. Similarly, for anomalous images when plotting histogram for distribution of density score the anomalous items should be far from normal item and the result shows that for anomalous item density score distribution is not far from the normal item in terms of distribution of density score.

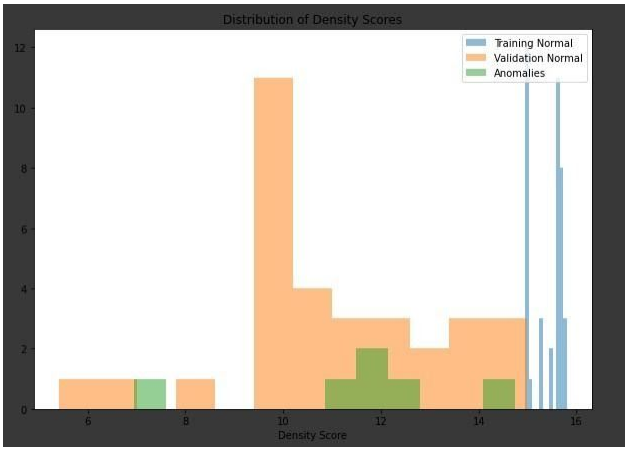

Figure 39 Density score distribution epoch 20

In the next phase epochs was increased to 100 but the training stopped after 48 epochs because validation did not improve.

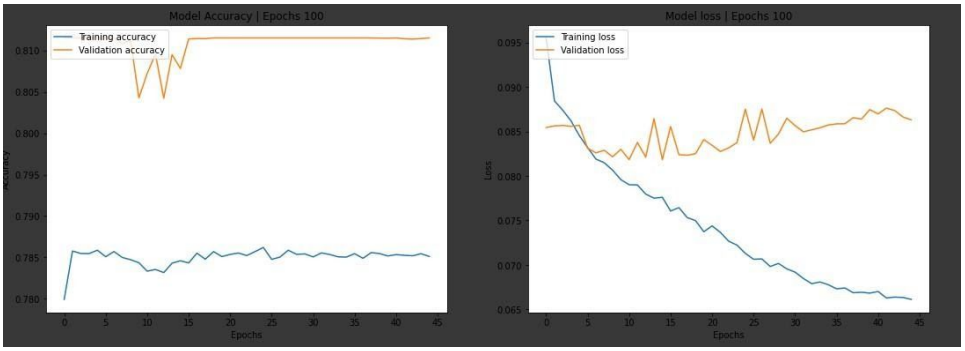

Figure 40 Model accuracy and loss in training and validation (Epochs 48)

The accuracy to detect anomalous item increased further when on increasing epochs to 48 but prediction does not provide higher level of confidence for detecting. For such experiments, it is necessary to provide accuracy to a level of 99.9% to trust the prediction. There was no improve in results after changing parameters other than epoch.

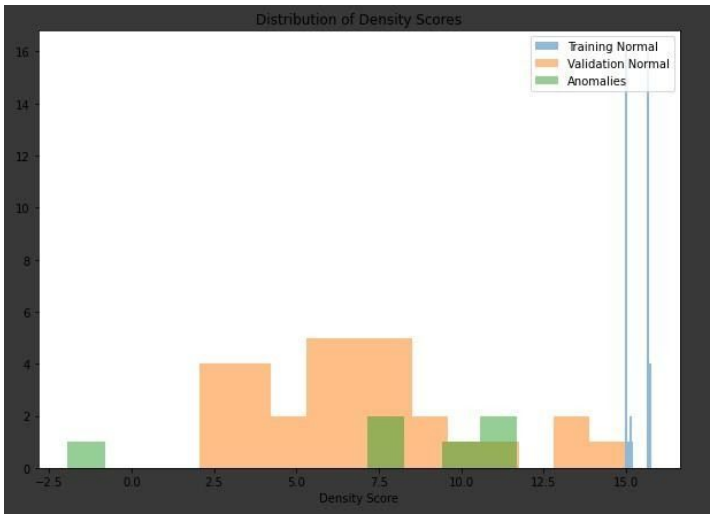

Figure 41 Density score distribution (epochs 48)

Generated images became less blur compared to the lesser number of epochs but did not improve significantly as shown in figure below.

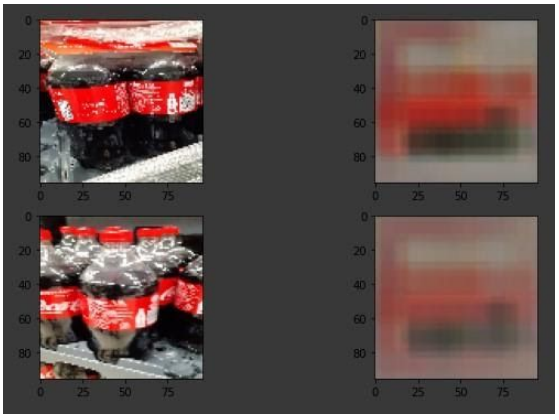

Figure 42 Example of real image vs generated image for coca cola (Epoch 48)

The experiment devised for this thesis was based on a simple Autoencoder model and comparing different autoencoder models were not examined. Lack of ample data combined with use of simple autoencoder that architecture resulted in the blurry generated image. This can be further investigated to find models and further experimentation with various autoencoder architecture that works better with small amount of training data.

Table 3 Result from experiment

Product	False positive	False negative	Correct prediction	Total samples
Coca cola	3	6	1	10
Apple	7	3	0	10
Washing liquid	1	9	0	10

Table 3 shows the result from the experiment. It was found that most of the results were either false positive or false negative. And only one item was predicted correctly. This shows that there is need for improvement in terms of amount of data and the selection of correct architecture.

6 Discussion

Result section of this thesis presents the finding and the outcome of the experiment that was conducted for this thesis. This section includes summery on the finding and study, discussion on the key findings and reflect on the research questions. Current limitations, future improvements and possible research areas are presented in the Limitation and future work section.

The purpose of this thesis was to first understand the possible usages of images anomalies detection in department store. Current available research work was studied to identify how anomalies detection can be used or has been used at department store. The second part was to conduct an anomaly detection experiment in department store to detect misplaced items in a shelve. The second part is a prototype of what can be done and uses Autoencoder combined Kernel Density Estimation (KDE) to measure the effectiveness of images anomalies detection in a department store.

The research questions for this thesis are listed below:

1. *What is the image-based anomaly detection methods?*
2. *What is the effectiveness of images anomalies detection in department store using Autoencoders combined with Kernel Density Estimation?*

Research question 1 consist of studying research works and available applications of images anomalies detection department. JYKDOK (tool provided by University of Jyväskylä to search electronic document), ResearchGate, online portals for searching research paper, and other online resources were used to find relevant research documents and books. Research question 3 included examining the effectiveness of anomalies detection in department store by devising a deep learning experiment and the conclusion of the findings are discussed in Conclusion section in detail.

7 Conclusion

This thesis consisted of study to understand deep learning principles that are useful for images anomalies detection. A sample prototype application was written to test the theory using autoencoder and kernel density estimation. Data collection was done using a smartphone and experiment data was stored in Google Drive. Experiment part of the thesis was done in Google Colab that provides runtime to conduct and document deep learning experiments. The ease of use of such platform as service helped to focus on the goal of the project without worrying about hardware and software needs as deep learning experiments can be intensive in terms of computing.

Despite not producing the expected result, the experiment conducted in this thesis provided a way to think how autoencoder with kernel density estimation can be used to detect anomalies in department store. The reflection from conducting and organizing the experiment was understanding how to manage a deep learning experiment, collect data and perform analysis on the performance of a deep learning experiment.

Images anomalies detection in department store is still a nascent field with massive potential. Some of the future work that can be performed are discussed in chapter 7.2.

7.1 Effectiveness of images anomalies detection in department store

The result obtained from this thesis experiment showed that it is possible to detect images anomalies misplaced item shelf or location in a department store using Autoencoder and Kernel Density Estimation method. However, the predicted confidence score is not reliable. The major bottle check of this experiment was collection of data. The other bottleneck is the categories of products that are available in the department store. For the purposed method to work in a department store context it would require training with large dataset and such hurdle can be overcome by using retail robots-based data collection. Using retail robots to collect normal samples would require making sure that the scanned shelves (product area) do not contain misplaced items during data collection. Using such robot would require minimal manpower compared to annotation each product area.

Another challenge to this approach is the dynamic nature of the department store and the products themselves. Some products are unique in color and size and when a product changes this is because the experiment lacked enough data for conducting the training due to data collection limitation posed as data collection was not done in collaboration with the department store.

7.2 Limitations and future work

This thesis research had limitations in terms of data collection and provides with only preliminary idea and prototype that can be extended in future to detect anomaly in department store regarding placement of products. Since the data collection which included taking pictures of items from shelves was done without any co-ordination with the department store. Anomalous situation had to created manually by placing item belonging to a different category and once it attracted the attention of the department store worker.

Anomaly in department store occurs in different situation, for example sudden rise in the number of customers, queueing at cash counter that is suddenly low or high, formation of a crowd in a certain area within a department store etc. can be detected by video anomaly. Future work could include use of video to detect anomaly using label. As shown in Figure 43 below; a camera mounted on a robot that can navigate floor using indoor navigation which then can send the video stream to the anomaly detection system which can detect the labels and make suggestions if an item is in a wrong place. The system can also suggest the correct shelf or category for the misplaced item. In figure the red box is an anomaly and using label detection it is possible to determine its correct place.

In addition to the label detection, text recognition on the product can be used to identify misplaced items combined with a security camera. Such experiment could include the security camera that is used by the department store. Text on the item could be recognized to better understand the correct placement of the product. For example, if a bottle of juice is place in the shelf which belongs to milk then enough context can be obtained by reading the text on the product label.

In fashion industry where labels are hidden then anomaly detection based on shape and size of an item can be devised. Zalando has published dataset namely fashion-mnist which consist of 60,000 examples and a test set of 10,000 examples having 28x28 pixel grayscale images (Fashion MNIST 2017). Such dataset available could be used in combination with video feed obtained from security cameras to detect when a piece of cloth is misplaced. Fashion-mnist dataset itself has a limitation because it consists of only 10 labels.

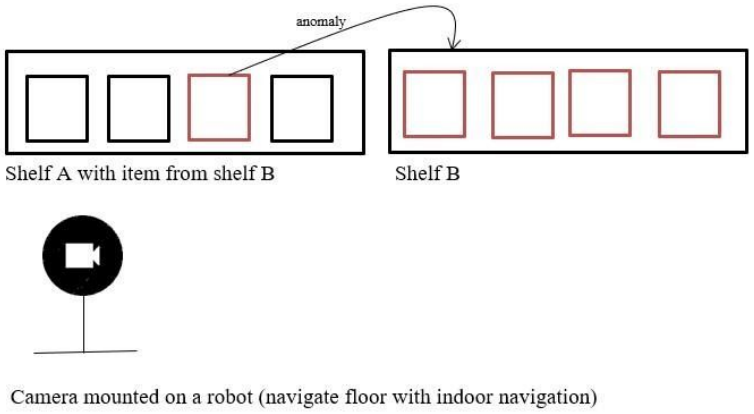

Figure 43 Future research work to extend the thesis work to detect anomalous product using video

Bibliography

Agostinelli, Forest , Matthew Hoffman, Peter Sadowski, and Pierre Baldi. 2015. *LEARNING ACTIVATION FUNCTIONS TO IMPROVE DEEP NEURAL NETWORK.* Research Paper, California: University of California.

"Automatic Realogram & Real Time Stock-out Detection." *www.ses-imagotag.com.* December. Accessed August 9, 2020. https://www.ses-imagotag.com/wp-content/uploads/2018/12/Product_Sheet_Shelf Watch_07122018.pdf.

Baddar, Sherenaz W. Al-Haj, Alessio Merio, and Mauro Migliardi. 2014. "Anomaly Detection in Computer Networks: A State-of-the-Art Review. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications." *JoWUA* 29-64.

Baldi, Pierre. 2012. "Autoencoders, Unsupervised Learning, and Deep Architectures." *Workshop on Unsupervised and Transfer Learning.* California: University of California. 1-2.

Bianchini, Monica, Marco Maggini, Lorenzo Sarti, and Franco Scarselli. 2005. "Recursive neural networks learn to localize faces."

Brown, Courtenay. 2020. *Stock market rises after Pfizer coronavirus vaccine news.* 10 November. Accessed November 10, 2020. https://www.axios.com/stock-market-pfizer-coronavirus-vaccine-c3c131d7-b46f-4df0-94c9-503d1dc906df.html.

Buscema, Massimo. 1998. "Back Propagation Neural Networks." *Substance Use & Misuse*, February.

Cai, Yuanqiang, Wen Longyin, Libo Zhang, Dawei Du, Weiqiang Wang, and Pengfei Zhu. 2020. *Rethinking Object Detection in Retail Stores.* Report, Institute of Software Chinese Academy of Sciences.

Chakravarty, Punarjay, Alan M Zhang, Ray Jarvis, and Lindsay Kleeman. 2020. *Anomaly Detection and Tracking for a Patrolling Robot.* Victoria, Australia: Monash University.

Chandola, Varun, Arindam Banerjee, and Vipin Kumar. 2009. *Anomaly Detection : A Survey.* Survey, Minnesota: University of Minnesota.

Cheng, Shi, and Chi-Man Pun. 2018. "Multi-scale Hierarchical Recurrent Neural Networks for Hyperspectral Image Classification." *Neurocomputing.*

Chollet, Franchois. 2016. *Building Autoencoders in Keras.* 14 5. Accessed 4 18, 2020. https://blog.keras.io/building-autoencoders-in-keras.html.

Chollet, Francois. 2016. *Building powerful image classification models using very little data.* 05 June. Accessed June 2020, 05. https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html.

Contrascere, Jared, and Matt Schwalm. 2019. *HOW GENERATIVE ADVERSARIAL NETWORKS CAN IMPACT BANKING.* 21 08. Accessed 05 2020, 21. https://financeandriskblog.accenture.com/analytics/how-generative-adversarial-networks-can-impact-banking.

Crispi, Alberto Taboada, Hichem Sahli, Maykel Orozoco Monteagudo, Denis Hernandez Pacheco, and Alexandar Falcon Ruiz. 2009. "ANOMALY DETECTION IN MEDICAL IMAGE ANALYSIS." *Handbook of Research on Advanced Techniques in Diagnostic Imaging and Biomedical Applications* 314-215.

Deng, Jia, Wei Dong, Richar Socher, Li-Jia Li, and Fei-Fei Li. 2014. "ImageNet: A Large-Scale Hierarchical Image Database." *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* Miami. 1-3.

Deshpande, Adit. 2019. *A Beginner's Guide To Understanding Convolutional Neural Networks.* Accessed 4 4, 2020.

https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/.

Donalek, Ciro. 2011. “Supervised and Unsupervised Learning.” 04. Accessed 03 25, 2020. http://www.astro.caltech.edu/~george/aybi199/Donalek_Classif.pdf.

E. Francesconi, P. Frasconi, M. Gori, S. MarinaiJ. J.Q. Sheng, G. Soda, A. Sperduti. 1997. “Logo Recognition by Recursive Neural Networks.” *International Workshop on Graphics Recognition.* 104-117.

Faggella, Daniel. 2020. *What is Machine Learning?* 26 02. Accessed 03 21, 2020. https://emerj.com/ai-glossary-terms/what-is-machine-learning/.

Fashion MNIST. 07 December. Accessed August 9, 2020. https://www.kaggle.com/zalando-research/fashionmnist.

Finland to invest in the future skills of Europeans – training one per cent of EU citizens in the basics of AI. 10 December. Accessed August 2020, 12. https://eu2019.fi/en/-/suomen-eu-puheenjohtajuuden-aloite-suomi-investoi-euroopp alaisten-tulevaisuustaitoihin-tavoitteena-kouluttaa-prosentti-eu-kansalaisista-tekoalyn-perus.

Géron, Aurélien. 2020. *Chapter 1. The Machine Learning Landscape.* Accessed August 10, 2020. https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/ch01.html.

Giles, Martin. 2018. *The GANfather: The man who’s given machines the gift of imagination.* 21 02. Accessed 05 2020, 10. https://www.technologyreview.com/2018/02/21/145289/the-ganfather-the-man-whos-given-machines-the-gift-of-imagination/.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. "Autoencoders." In *Deep Learning*, 499-523. MIT Press.

Goyal, Mohit, Rajan Goyal, and Brejesh Lall. 2019. *Learning Activation Functions: A new paradigm for understanding Neural Networks.* Research, Delhi: IIT Delhi.

Gu, Jiuxiang, Zhenhua Wang, Kuen Jason, Lianyang Ma, Amir Shahroudyb, Bing Shuai, Ting Liu, et al. 2017. *Recent Advances in Convolutional Neural Networks.* Singapore: Nanyang Technological University.

Gurney, Kevin. 1997. *An introduction to neural networks.* London: UCL Press Limited.

Henver, Alan, Salvatore T. March, Jinsoo Park, and Sudha Ram. 2004. "Design Science in Information Systems Research." *MIS Quaterly*, March: 75-105.

Hu, Wei, Yangyu Huang, Li Wei, Fan Zhang, and Hengchao Li. 2014. "Deep Convolutional Neural Networks for." *Journal of Sensors* 1-2.

Hurwitz, Judith, and Kirsch Daniel. 2018. *Machine Learning for Dummies, IBM Limited Edition.* New Jersey: John Willy & Sons, Inc.

"IBM100 - Deep Blue." *Icons of Progress.* Accessed November 11, 2020. https://www.ibm.com/ibm/history/ibm100/us/en/icons/deepblue/.

Jain, Vandit. 2019. *Everything you need to know about "Activation Functions" in Deep learning models.* 30 12. Accessed 03 12, 12. https://towardsdatascience.com/everything-you-need-to-know-about-activation-functions-in-deep-learning-models-84ba9f82c253.

Joshi, Chandrashekhar, and Swagat Kumar. 2014. *Robotics-as-a-Service: Transforming the Future of Retail.* Project Report, January.

Khoshgoftaar, Taghi , Maryam M Najafabadi, Flavio Villanustre, and Naeem Seliya. 2015. "Deep learning applications and challenges in big data analytics." *Journal Of Big Data* 2-5.

Lefkowitz, Melanie. 2019. *Professor's perceptron paved the way for AI – 60 years too soon.* 25 September. Accessed November 15, 2020.

https://news.cornell.edu/stories/2019/09/professors-perceptron-paved-way-ai-60-years-too-soon.

Li, Deng, and Yu Dong. 2014. "Deep Learning Methods and Applications." *https://www.microsoft.com/en-us/research/.* Accessed 03 21, 2020. https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/DeepLearning-NowPublishing-Vol7-SIG-039.pdf.

Maad, Mijwil M., Adam Esen, and Aysar Alsaadi. 2019. *Overview of Neural Networks.* Baghdad: Baghdad College of Economics Sciences University.

Maimon, Ode, and Lior Rokach. 2020. "INTRODUCTION TO SUPERVISED METHODS." Accessed 4 4, 23. http://www.ise.bgu.ac.il/faculty/liorr/hbchap8.pdf.

Markou, Markos, and Sameer Singh. 2003. *Noveltydetection: a review—part 1: statistical approaches.* University of Exeter.

Marr, Bernard. 2019. *The Amazing Ways YouTube Uses Artificial Intelligence And Machine Learning.* 23 August. Accessed November 2020, 11. https://www.forbes.com/sites/bernardmarr/2019/08/23/the-amazing-ways-youtube-uses-artificial-intelligence-and-machine-learning/?sh=365768b95852.

Minhas, Manpreet Singh, and John Zelek. 2019. *Anomaly Detection in Images.* Report, Waterloo: University of Waterloo.

Moolayil, Jojo John. 2019. *A Layman's Guide to Deep Neural Networks.* 24 July. Accessed November 15, 2020. https://towardsdatascience.com/a-laymans-guide-to-deep-neural-networks-ddcea24847fb.

Moore's Law. Accessed November 11, 10. http://www.mooreslaw.org/.

Mou, Lichao, Pedram Ghamisi, and Xiao Xiang. 2017. "Deep Recurrent Neural Networks for." *IEEE Transactions on Geoscience and Remote Sensing* 3639.

Najafabadi, Maryam, Flavio Villanustre, Taghi M Khoshgoftaar, Naeem Seliya, Randall Wald, and Edin Muharemagic. 2015. “Deep learning applications and challenges in.” *Journal of Big Data* 2-3.

Nguyen, Thanh Thi, Cuong M. Nguyen, Dung Tien Nguyen, Duc Thanh Nguyen, and Saeid Nahavandi. 2020. *Deep Learning for Deepfakes Creation and Detection: A Survey.* IEEE, 1-5.

Pan, Sinno Jialon, and Qiang Yang. 2010. “A Survey on Transfer Learning.” *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING* 1345-1359.

Pang, Guansong, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. 2020. “Deep Learning for Anomaly Detection: A Review.” Adelaide.

Perceptrons and Multi-Layer Perceptrons: The Artificial Neuron at the Core of Deep Learning. Accessed November 15, 2020. https://missinglink.ai/guides/neural-network-concepts/perceptrons-and-multi-layer-perceptrons-the-artificial-neuron-at-the-core-of-deep-learning/.

Phan, Nhathai, Deijing Dou, Birgitte Piniewski, and David Kil. 2016. *A deep learning approach for human behavior prediction with explanations in health social networks: social restricted Boltzmann machine (SRBM+).* Manuscript, US National Library of Medicine, National Institutes of Health.

Puthige, Isha. 2020. *Beginners Guide To Transfer Learning with an example using VGG16.* 13 November. Accessed November 15, 2020. https://medium.com/@ishaputhige/beginners-guide-to-transfer-learning-with-simple-example-using-vgg16-61040c095ab9.

Rosebrock, Adrian. 2020. “Anomaly detection with Keras, TensorFlow, and Deep Learning.” *https://www.pyimagesearch.com.* 2 3. Accessed 4 12, 2020. https://www.pyimagesearch.com/2020/03/02/anomaly-detection-with-keras-tensorflow-and-deep-learning/.

Ruban, Timon. 2018. *The Deep Learning Toolset — An Overview.* 18 November. Accessed July 12, 22. https://medium.com/luminovo/the-deep-learning-toolset-an-overview-b71756016c06.

Sadr, Alireza Vafaei, Bruce A. Bassett, and M Kunz. 2019. "A Flexible Framework for Anomaly Detection via." Geneva.

Sarkar, Dipajan (DJ). 2018. *A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep Learning.* 15 11. Accessed 04 2020, 12. https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a.

Schmidhuber, Jürgen. 2015. *who invented backpropagation?* Accessed 4 1, 2020. http://people.idsia.ch/~juergen/who-invented-backpropagation.html.

Shuai, Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. "Deep Learning based Recommender System: A Survey and New Perspectives." *ACM Computing Surveys.*

Singh, Hardeep, Ashley N D Meyer, and Eric J Thomas. 2014. "The frequency of diagnostic errors in outpatient care: estimations from three large observational studies involving US adult populations." *BMJ Quality & Safety* 727-731.

Socher, Richard, Christopher D. Manning, and Andrew Y. Ng. n.d. *Learning Continuous Phrase Representations and.* Department of Computer Science, Stanford University.

Solteq Retail Robot. Accessed November 15, 2020. https://www.solteq.com/en/robotics/solteq-retail-robot.

Tai, Y-Chin, Cheng-wei Chan, and Jane Yung-jen Hsu. 2010. *Automatic Road Anomaly Detection Using Smart Mobile Device.* Conference Paper, Taipei: National Taiwan University.

Thanh, Nguyen Thi, Nguyen M. Cuong, Gnuyen Tien Dung, Thanh Nguyen Duc, and Nahavandi Saeid. 2019. “Deep Learning for Deepfakes Creation and Detection.” 09. Accessed 02 20, 2020. https://arxiv.org/pdf/1909.11573.pdf.

Turing, Alan. 1950. “Computing machinery and intelligence.” 1-5.

Urban, Sebastian. 2017. *Neural Network Architectures.* PhD Thesis, Munich: Technical University Munich.

Vondrick, Karl, Hamid Pirsiavash, and Antonio Torralba. 2016. “Generating Videos with Scene Dynamics.” *29th Conference on Neural Information Processing Systems (NIPS 2016).* Barcelona: MIT, University of Maryland Baltimore County.

Wagner, Janet. 2014. “Deep Learning: 6 Real World Use Cases.” *https://www.jwtechwriter.com.* Accessed 03 20, 2020. https://www.jwtechwriter.com/pdfs/deep-learning-6-use-cases.pdf.

Węglarczyk, Stanisław. 2018. “Kernel density estimation and its application.” *ITM Web of Conferences. 23. 00037. 10.1051/itmconf/20182300037.* Warsaw.

Wells, Jude. 2020. *Image Anomaly Detection / Novelty Detection Using Convolutional Auto Encoders In Keras & Tensorflow 2.0.* 20 January. Accessed November 2020, 12. https://medium.com/@judewells/image-anomaly-detection-novelty-detection-using-convolutional-auto-encoders-in-keras-1c31321c10f2.

Wiggers, Kyle. 2018. *DeepMind’s AlphaZero beats state-of-the-art chess and shogi game engines.* 6 December. Accessed November 2020, 11. https://venturebeat.com/2018/12/06/google-deepmind-alphazero-chess-shogi-go/.

Wolff, Rachel. 2020. *11 Natural Language Processing (NLP) Applications in Business.* 20 May. Accessed November 2020, 21. https://monkeylearn.com/blog/natural-language-processing-applications/.

Yousoufi-Azar, Mahmood, Vijay Varadharajan, Len Hamey, and Uday Tupakula. 2018. *Autoencoder-based Feature Learning for Cyber Seucity Applications.* Sydney: Department of Computing, Faculty of Science and Engineering.

Yse, Diego Lopez. 2019. *Your Guide to Natural Language Processing (NLP).* 15 January. Accessed November 2020, 12. https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1.

Zhai, Hao. 2016. "Research on Image Recognition Based on Deep Learning Technology." *4th International Conference on Advanced Materials and Information Technology Processing (AMITP 2016)* . Tianjin: School of Computer and Communication Engineering, Tianjin University of Technology. 266-268.

Zhao, Liang. 2007. *Event Prediction in the Big Data Era: A Systematic Survey.* Report, Atlanta: Emory University.

Appendices

A Reading training data

```
import os
import keras
import random
import matplotlib.pyplot as plt
import numpy as np
from keras.models import Sequential
from keras.layers import Dense, Activation, Flatten, Input
from keras.layers import Conv2D, MaxPooling2D, UpSampling2D
from keras import backend as K
from keras.preprocessing.image import ImageDataGenerator, arr
ay_to_img, img_to_array, load_img
from PIL import Image, ImageChops
from sklearn.neighbors import KernelDensity
from keras.callbacks import Callback
from keras.callbacks import History


# mount google drive
from google.colab import drive
drive.mount('/content/drive')

drive_folder_location='/content/drive/My Drive/Experiment/'


epochs = 50
model_filepath = drive_folder_location+'models/saved_best_mod
el_epochs_'+str(epochs)+'.keras'
```

```
# create generators


batch_size = 32

image_generator = ImageDataGenerator(rescale=1./255, data_for
mat='channels_last')
train_generator = image_generator.flow_from_directory(
    drive_folder_location+'training_images',
    target_size=(96, 96),
    batch_size=batch_size,
    class_mode='input'
    )

validation_generator = image_generator.flow_from_directory(
    drive_folder_location+'test_images',
    target_size=(96, 96),
    batch_size=batch_size,
    class_mode='input'
    )

anomaly_generator = image_generator.flow_from_directory(
    drive_folder_location+'anomalous_data',
    target_size=(96, 96),
    batch_size=batch_size,
    class_mode='input'
    )
```

B Training model

```
model = Sequential()

model.add(Conv2D(16, (3, 3), padding='same',activation='relu'
, input_shape=(96, 96, 3)))
model.add(MaxPooling2D(pool_size=(4,4), padding='same')) # us
ing pool_size (4,4) makes the layer 4x smaller in height and
width

model.add(Conv2D(8,(3, 3),activation='relu',  padding='same')
)
model.add(MaxPooling2D(pool_size=(4,4), padding='same'))

model.add(Conv2D(3,(3, 3),activation='relu',  padding='same')
)
model.add(MaxPooling2D(pool_size=(2,2), padding='same'))

#-------------------------
model.add(Conv2D(3,(3, 3),activation='relu',  padding='same')
)
model.add(UpSampling2D((2, 2)))

model.add(Conv2D(8,(3, 3),activation='relu',  padding='same')
)
model.add(UpSampling2D((4, 4)))

model.add(Conv2D(16,(3, 3),activation='relu',  padding='same'
))
model.add(UpSampling2D((4, 4)))
```

```
model.add(Conv2D(3,(3, 3), activation='sigmoid', padding='sam
e'))
#-------------------------

model.summary()

# Compile the model
model.compile(optimizer='adadelta', loss='mean_squared_error'
,metrics=['acc','mean_squared_error'])

# Training the model
es = keras.callbacks.EarlyStopping(monitor='val_loss', mode='
min', verbose=1, patience=30) # Early stopping (stops trainin
g when validation doesn't improve for {patience} epochs)
history = History()
save_best = keras.callbacks.ModelCheckpoint(model_filepath, m
onitor='val_loss', save_best_only=True, mode='min') # Saves t
he best version of the model to disk (as measured on the vali
dation data set)

# save model
plot_model(model, to_file='model_plot.png', show_shapes=True,
 show_layer_names=True)
```

C Check anomaly function

```
def check_anomaly(img_path):
```

```
    density_threshold = 0 # This threshold was chosen based o
n looking at the distribution of the density scores of the no
rmal class (validation set)
    reconstruction_error_threshold = 0.04 # This threshold wa
s chosen based on looking at the distribution of reconstructi
on errors of the normal class
    img  = Image.open(img_path)
    img = np.array(img.resize((96,96), Image.ANTIALIAS))
    img = img / 255
    encoded_img = encoder_replica.predict([[img]]) # Create a
 compressed version of the image using the encoder
    encoded_img = [np.reshape(img, (27)) for img in encoded_i
mg] # Flatten the compressed image
    density = kde.score_samples(encoded_img)[0] # get a densi
ty score for the new image
#     print(f'density: {density}')
    reconstruction = model.predict([[img]])
    reconstruction_error = model.evaluate([reconstruction],[[
img]], batch_size = 1)
    print(f'reconstruction_error: {reconstruction_error}')
    if density < density_threshold or reconstruction_error[2]
 > reconstruction_error_threshold:
        return True
    else:
        return False
```