Integrating Structured and Unstructured Data for Imbalanced Classification Using Meat-Cut Images

Satya Prakash BTech, MSc

Thesis Submitted for the Award of MSc.

School of Computing

Dublin City University

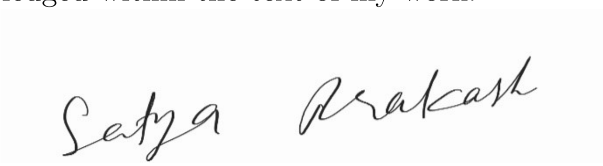
Supervisors

Dr. Andrew McCarren and Dr. Paul M. Clarke

January 2024

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Master of Science is entirely my own work, and that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: 

ID No.: 19210200

Date: December 19, 2023

Acknowledgements

I would like to express my deepest gratitude to all those who have contributed to the completion of this thesis. Their support, guidance, and encouragement have been invaluable throughout this journey.

First and foremost, I am profoundly grateful to my supervisors, Dr. Andrew McCarren, and Dr. Paul M. Clarke, for their unwavering support, expert guidance, and continuous encouragement. Their wealth of knowledge, insightful feedback, and dedication to my academic growth have been instrumental in shaping this research. I am truly fortunate to have had the opportunity to work under their mentorship.

I am indebted to the faculty and staff of DCU, whose commitment to academic excellence and the pursuit of knowledge has provided me with a stimulating and conducive learning environment. The resources, facilities, and opportunities made available to me have been instrumental in the successful completion of this thesis.

My deepest gratitude goes to my family for their unwavering love, encouragement, and belief in my abilities. Their constant support, understanding, and sacrifices have been the pillar of strength that has propelled me forward in times of doubt and challenges.

I am immensely grateful to my friends and classmates who have provided me with companionship, moral support, and a sense of camaraderie throughout this academic journey. Their shared experiences, intellectual discussions, and moments of levity have made this undertaking more meaningful and enjoyable. I am grateful to the numerous scholars, researchers, and authors whose seminal works have laid the foundation for this study. Their contributions have provided a robust framework upon which this research has been built, and I am indebted to their intellectual brilliance.

Lastly, I would like to acknowledge the financial support provided by Science Foundation Ireland. Their investment in my education has enabled me to pursue this research and bring it to fruition.

To all those whose names may not be mentioned but have played a role in shaping this thesis, whether through their encouragement, discussions, or assistance, I offer my sincere gratitude.

Completing this thesis would not have been possible without the collective support and contributions of these individuals and institutions. I am humbled and privileged to have been surrounded by such a remarkable network of support.

Thank you all for being an integral part of this incredible journey.

List of Publications

Prakash, S., Berry, D.P., Roantree, M., Onibonoje, O., Gualano, L., Scriney, M. and McCarren, A., 2021. Using artificial intelligence to automate meat cut identification from the semimembranosus muscle on beef boning lines. Journal of Animal Science, 99(12), p.skab319.

Contents

List of Abbreviations

AI Artificial Intelligence

CNN Convolutional Neural Network

VPA Virtual Personal Assistant

MII Meat Industry Ireland

CSO Central Statistics Office

LR Logistic Regression

PLS-DA partial least square discriminant analysis

LDA linear discriminant analysis

DT Decision Tree

KNN K-Nearest Neighbor

ANN Artificial Neural Network

SVM Support Vector Machine

HSI hyperspectral images

YOLO You Only Look Once

MDS Multidimensional Scaling

SMOTE Synthetic Minority Oversampling Technique

DATAS Deductive Analytics for Tomorrow's Agri Sector

SOL Start of Line

EOL End of Line

MLR Multinomial logistic regression

DTC Decision tree classifiers

ResNet Residual Network

List of Tables

List of Figures

Integrating Structured and Unstructured Data for Imbalanced Classification Using Meat-Cut Images

Satya Prakash

Abstract

The identification of different meat cuts for labeling and quality control on production lines is still largely a manual process. As a result, it is a labor-intensive exercise with the potential for not only error but also bacterial cross-contamination. Artificial intelligence is used in many disciplines to identify objects within images, but these approaches usually require a considerable volume of images for training and validation. The objective of this study was to integrate structured and unstructured data to identify five different meat cuts from images and weights collected by a trained operator within the working environment of a commercial Irish beef plant. The dataset for one of the products exhibited sparsity, resulting in an imbalanced distribution. To rectify this issue, image augmentation techniques were employed to tackle the inherent imbalance within the dataset. Individual cut images and weights from 7,987 meats cuts extracted from semimembranosus muscles (i.e., Topside muscle), post editing, were available. A variety of classical neural networks and a novel Ensemble machine learning approaches were then tasked with identifying each individual meat cut; performance of the approaches was dictated by accuracy (the percentage of correct predictions), precision (the ratio of correctly predicted objects relative to the number of objects identified as positive), and recall (also known as true positive rate or sensitivity). A novel Ensemble approach outperformed a selection of classical neural networks including convolutional neural network and residual network. The accuracy, precision, and recall for the novel Ensemble method were 99.13%, 99.00%, and 98.00%, respectively, while that of the next best method were 98.00%, 98.00%, and 95.00%, respectively. The Ensemble approach, which requires

relatively few gold-standard measures, can readily be deployed under normal abattoir conditions; the strategy could also be evaluated in the cuts from other primals or indeed other species such as lamb, chicken, or pork.

Chapter 1

Introduction

The study of artificial intelligence (AI) has philosophical, mathematical, and computer-scientific foundations. The development of artificial intelligence may be traced back to the period when scientists and philosophers were enthralled by the idea of building intelligent machines. However, the development of electronic computers in the 1940s marked the beginning of the contemporary history of AI (Russell 2010).

British mathematician Alan Turing made one of the earliest important contributions to AI when he created the idea of a universal computer that could carry out any computation that a person could. He also proposed the Turing Test, which determines if a machine can display intellect comparable to a person's. Turing's contributions laid the groundwork for AI research, and his theories on computing and AI are still relevant today (Turing 1950; Russell 2010).

The term "Artificial Intelligence" was first used by a group of scientists that included John McCarthy, Marvin Minsky, and Claude Shannon in the 1950s (Press 2022). They then started looking into how to build computers that could think and reason like humans. They created the Logic Theorist and the General Problem Solver, two early AI algorithms. These early programs had great potential but were constrained by the processing capability at the time (Russell 2010).

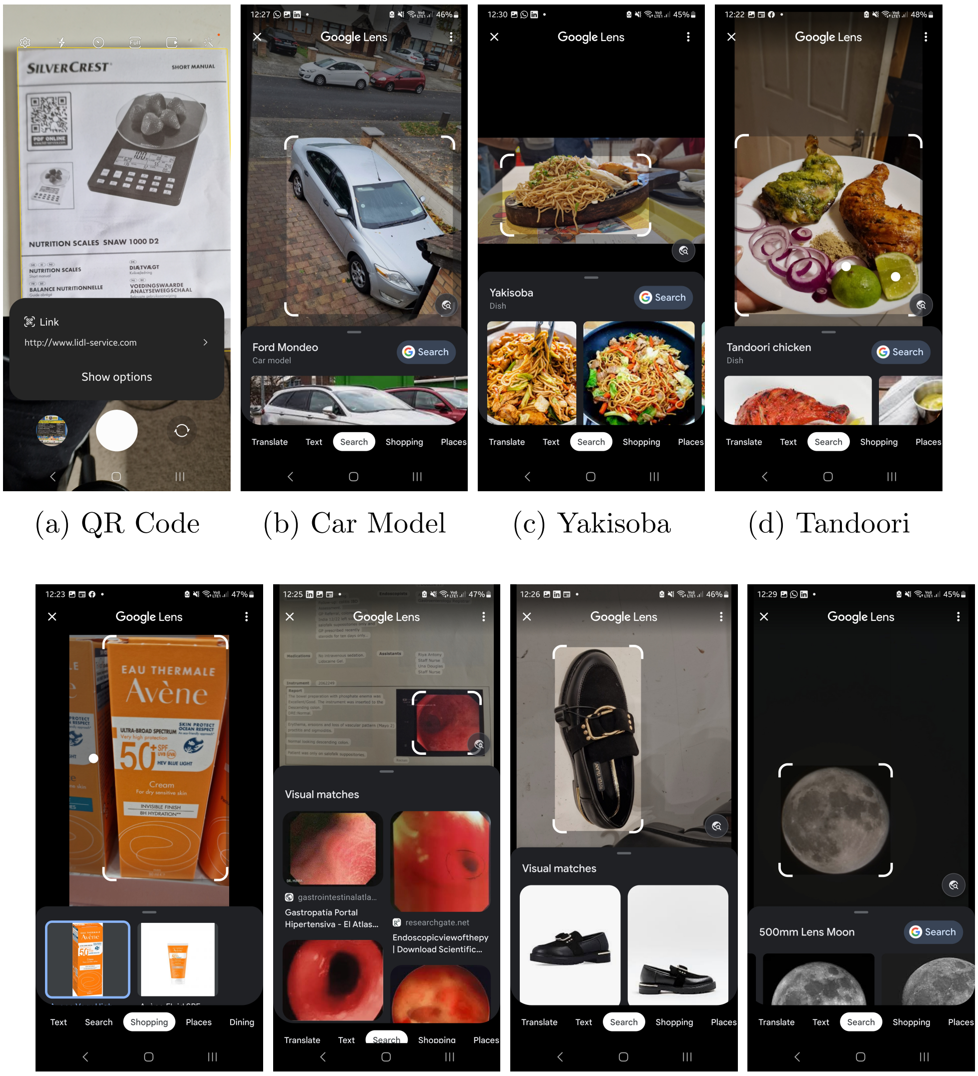
AI research started to flourish in the 1960s and 1970s when substantial advancements were achieved in fields like computer vision and natural language processing. At Stanford University in 1965, Edward Feigenbaum and Joshua Lederberg created the first expert system, known as Dendral. Expert systems were utilized extensively in business and government because they were created to emulate a human expert's decision-making abilities in a given subject (J. McCarthy 2007).

However, the hype surrounding AI led to unrealistic expectations, and progress slowed down in the 1980s. Funding for AI research was reduced, and many researchers shifted their focus to other areas.

AI research began to focus on more useful applications in the 1990s, such as machine learning and expert systems. Computers can now spot patterns and make predictions thanks to the development of machine learning algorithms that can learn from data. The World Wide Web's introduction during this time period was one of the most important advancements in AI since it made it possible for researchers to more easily exchange data and cooperate.

In recent years, AI has experienced a resurgence, fueled by advances in computing power, big data, and deep learning. AI technologies are now being used in various applications, from self-driving cars and speech recognition systems to medical diagnosis and financial analysis (Russell 2010).

One of the applications of AI is picture analysis using a camera. When a picture is taken from any smartphone camera, the images are automatically labeled based on the objects in that picture. This is quite an interesting feature of a smartphone that uses the concept of artificial intelligence and in particular deep learning to classify images. Figure 1.1 shows how Google Lens identifies various objects in images including a QR code, a car model, Yakisoba, tandoori chicken, sunscreen, a colon image from a colonoscopy report, a shoe, and the moon. Google uses deep learning models based on Convolutional Neural Networks (CNNs) to classify various images. Their image classification models have been trained on massive datasets containing millions of images, covering thousands of different object categories. They have achieved state-of-the-art performance on standard image classification benchmarks, demonstrating the effectiveness of deep learning approaches for image classification tasks.



(a) QR Code (b) Car Model (c) Yakisoba (d) Tandoori

(e) Sunscreen (f) Colon (g) Shoe (h) Moon

Figure 1.1: Object Detection in Google Photos using Google Lens.

1.1 Prevalence of Artificial Intelligence in Society

AI is becoming increasingly prevalent in society, and its use is expanding across a wide range of fields and industries. Here are some examples of how AI is being used in society.

1.1.1 Virtual Personal Assistants

Digital assistants called Virtual Personal helpers (VPAs) employ artificial intelligence to comprehend and carry out spoken commands in natural language. Siri is an Apple-created virtual assistant that can do a variety of things, like setting reminders, sending messages, placing calls, and conducting online searches. The Google Assistant is a virtual assistant that can respond to queries, send reminders, manage smart home appliances, and offer tailored advice based on user behavior. Amazon created Alexa, a virtual assistant that can operate smart home appliances, play music, respond to inquiries, and offer news updates. Microsoft created the virtual assistant Cortana, which can do things like send emails, make reminders, and provide weather information. Bixby is a virtual assistant developed by Samsung that can control smart home devices, perform web searches, and send text messages (Inc. 2011; LLC 2016; Coates and Amordeluso 2019; Group 2023; Corporation 2014).

1.1.2 Social Media

AI is employed in a number of ways and is becoming more and more significant in social media. To deliver tailored information and suggestions, AI systems examine user behavior, preferences, and interests. Social media networks benefit from increased engagement and user retention as a result. It is used to identify and delete offensive material, hate speech, and fake news. This keeps the environment around users secure and healthy. Ads are targeted to consumers based on their interests, demographics, and behavior using AI algorithms. This makes it easier and more effective for advertising to reach their target market. It is used to assess the tone of user-generated material, including reviews and comments. This enables firms and brands to comprehend the viewpoints and preferences of their target market and modify their strategic plans accordingly. AI-driven chatbots are used in social media messaging to perform basic transactions, offer customer support, and respond to commonly asked inquiries. AI is being used to identify and classify pictures and videos (Rai 2020; Granik and Mesyura 2017; Ozbay and Alatas 2020), making it simpler to find and arrange material.

1.1.3 Healthcare

AI is being used in various ways in healthcare, from diagnosing diseases to developing new drugs. AI algorithms are being used to analyze medical images such as X-rays, CT scans, and MRIs. These algorithms can help detect abnormalities that may be difficult for human radiologists to spot, leading to faster and more accurate diagnoses. AI is being used to develop new drugs by analyzing large amounts of data and identifying potential drug candidates. This can speed up the drug development process and reduce costs. Also, it is being used to predict and prevent diseases by analyzing patient data and identifying patterns and risk factors. This can help doctors to diagnose diseases earlier and develop personalized treatment plans for patients. AI-powered virtual assistants are being used to help patients manage their health by providing personalized recommendations and reminders. These assistants can also answer questions and provide support for patients with chronic conditions. AI-enabled robots are being used in surgery to assist surgeons with tasks such as suturing, cutting, and manipulating tissues. This can improve the precision and accuracy of surgical procedures and reduce the risk of complications (Panch, Mattie, and Celi 2019; F. Jiang et al. 2017; K.-H. Yu, Beam, and Kohane 2018).

1.2 Artificial Intelligence in Agriculture and Meat Plants

AI has the potential to revolutionize agriculture and meat processing plants by increasing efficiency, reducing costs, and improving sustainability. Here are some ways AI is being used in these industries:

- Precision Agriculture: AI technologies can be used to optimize the use of water, fertilizer, and pesticides, based on real-time data collected from sensors and drones. This can help reduce waste and increase yields (Patrıcio and Rieder 2018; Selvaraj et al. 2019).

- Crop Monitoring: AI can be used to monitor crops and detect issues such as diseases, pests, or nutrient deficiencies before they become visible to the human eye. This can help farmers take action before the crops are damaged, leading to better yields and quality (Singh, S. Srivastava, and Mishra 2020).

- Livestock Management: AI technologies can be used to monitor the health and behavior of livestock, such as detecting early signs of illness or stress. This can help farmers take proactive measures to prevent disease outbreaks, reduce the need for antibiotics, and improve animal welfare (Neethirajan and Kemp 2021; Fricke et al. 2014).

- Meat Processing: AI can be used to automate various aspects of meat processing, such as sorting, grading, and packaging. This can help reduce labor costs, improve quality control, and increase efficiency.

- Supply Chain Management: AI can be used to optimize logistics and supply chain management, from predicting demand to tracking inventory and shipping. This can help reduce waste, minimize transportation costs, and improve sustainability (Zawish et al. 2022; Nayal et al. 2022).

1.3 Challenges for Irish Meat Industry

The agri-food sector includes primary production in farming, fishing, and forestry, and the processing and manufacture of food, beverages, and wood products. It is Ireland's oldest and largest indigenous sector and reaches out to every corner of the country and across our seas. In 2021, the sector employed 170,400 people, representing 7.1% of the total workforce across 135,000 farms, 2,000 fishing vessels aquaculture sites and some 2,000 food production and beverage enterprises. The sector is responsible for 4.5 million hectares of agricultural land, 770,000 hectares of forestry and producing close to 10% of Ireland's exports each year (Agriculture 2022).

Up to 90% of Ireland's food production is thought to be exported. Over 180 nations around the world imported Irish food and drink in 2021. Exports of agri-food products reached a new high of €15.4 billion, up from €10.2 billion in 2012 and representing a 50% rise over the previous ten years. Irish merchandise exports made up 9.4% of agri-food exports, which generated a €5.6 billion trade surplus (Agriculture 2022).

Increasing population and rising affluence in developing nations are predicted to contribute to a 1.4% annual increase in global meat consumption. The world will need to import an additional 3.4 million tonnes of beef to make up the difference between domestic demand and production in several nations. In the EU meat markets, sustainability is anticipated to become increasingly important for both consumers and farmers. By 2031, it is anticipated that per capita meat consumption in the EU would slightly decrease to 67 kg as consumers' environmental awareness, health concerns, and convenience patterns shift (Office 2022).

In 2021, the supply of Total Meat in Ireland experienced a decline of 37,000 tonnes (-3%), reaching a total of 1.423 million tonnes (Office 2022). Among this total, Beef & Veal accounted for 637,000 tonnes (45%). Slaughterings also saw a decrease, falling by 27,000 tonnes (-2%), with the exception of Pigs, which showed an increase of 15,000 tonnes (+5%).

The largest decline in slaughterings was observed in the Beef & Veal category, which decreased by 39,000 tonnes (-6%) to 595,000 tonnes. Net exports, which represent the difference between exports and imports, fell by 47,000 tonnes (-6%) to 707,000 tonnes. Within this reduction, Beef & Veal accounted for 40,000 tonnes.

Among the different categories of meat, Poultry Meat was the only one with net imports, as imports exceeded exports by 19,000kg. In terms of self-sufficiency, Ireland's overall self-sufficiency in Total Meat decreased by 19% to 267% in 2021 compared to 2020.

However, there were variations in self-sufficiency rates for specific types of meat. Pig Meat saw an increase of 8% to 240% in self-sufficiency, while Sheep Meat experienced a 3% rise to 361%. On the other hand, self-sufficiency rates decreased significantly for Beef & Veal by 51% to 661% and for Poultry Meat from 100% in 2020 to 90% in 2021.

The meat industry also generates significant exports, accounting for approximately 30% of Ireland's total food and drink exports. Ireland is particularly renowned for its beef exports, which are recognized globally for their high quality and safety standards. Moreover, the meat industry has a significant impact on rural economies, where many meat processing plants are located. The industry provides a significant source of income for farmers who supply the raw materials for meat production.

Access to a skilled and experienced workforce is fundamental to businesses that depend on human intervention in their production processes. The meat industry is one such sector, and this was highlighted by the levels of absenteeism during the coronavirus disease 2019 (COVID-19) restrictions. Processes such as meat cutting, fat determination, and meat deboning have been partially automated (Bostian et al. 1985; Umino et al. 2011). However, the labeling and identification of meat cuts still require a substantial amount of human intervention and manual handling. This can incur additional labor costs as well as being a source of error and potential microbiological contamination (Choi et al. 2013).

1.3.1 Maintaining Hygiene

Food safety and animal welfare have been ongoing issues in the meat processing industry globally. In Ireland, there have been reports of meat plants failing to comply with hygiene standards, resulting in product recalls and concerns over the safety of meat products. One of the main concerns is the potential for cross-contamination. This can occur when workers handle meat products without proper hygiene practices, such as washing their hands and wearing gloves, hairnets, and other protective clothing. This can lead to the transfer of bacteria and other contaminants from workers to meat products, which can cause foodborne illnesses if consumed by humans.

Cross-contamination can have a significant impact on the shelf life of meat products. Cross-contamination occurs when harmful bacteria or microorganisms are transferred from one surface or food item to another. In the case of meat products, if contaminated meat comes into contact with uncontaminated meat, it can cause the spread of harmful bacteria and pathogens, such as Salmonella, E. coli, or Listeria. These microorganisms can cause spoilage and shorten the shelf life of meat products, reducing their safety for consumption. Additionally, the growth of bacteria can result in the production of harmful toxins that can lead to foodborne illnesses when consumed by humans.

In addition, the presence of humans in meat plants can also increase the risk of spreading diseases among workers. This was evident during the COVID-19 pandemic, where meat processing plants were identified as high-risk environments due to the close proximity of workers and the potential for the virus to spread through respiratory droplets. Overall, while the presence of humans in meat processing plants can present hygiene challenges, it is possible to mitigate these risks through proper protocols, training, and the use of technology.

1.3.2 Labor Shortage

The robust economic growth in Ireland in the past few years resulted in a drop in unemployment rates below 5%, causing significant recruitment challenges in various industries, including meat processing. Meat Industry Ireland (MII) members prioritize local recruitment, collaborating with government departments to identify suitable candidates from the live register. If local recruitment efforts fail, they extend their search throughout the country and across the European Union to hire workers with the necessary skills (Ibec 2022; Accounts 2022).

The meat processing industry has been struggling to recruit the required workforce in recent years, leading to a significant labor shortage. To tackle this issue, meat processors have turned to the Employment Permit system implemented by the government, which allows them to hire skilled workers from outside the EU. The employment permit scheme has allowed more than 3,000 workers from outside the EU to work for Irish meat industries. Under the Employment Permit system, employees are authorized to work in Ireland for a maximum of two years, with the option to extend their permits upon expiration. All workers, including permit holders, are protected by Irish employment laws and are entitled to the same employment rights and benefits. Furthermore, the regulations of the permit scheme prescribe specific employer obligations, such as ensuring a safe and healthy work environment and complying with employment, health and safety, and anti-discrimination laws. At present, there are 50% Irish, 30% EU/EEA (excluding Ireland), and 20% non-EU/EEA (via the Employment Permit system) in meat industries.

In the meat industry, a beef primal cut refers to a large section of meat that is initially separated from the carcass during the butchering process. Primal cuts are typically divided based on the anatomical structure of the animal and are further broken down into smaller retail cuts that are found at the grocery store or butcher shop. (Warriss 2010; Underly 2011)

Primal boning lines are a typical example of where multiple operators simultaneously work on a range of meat cuts. Each cut will eventually arrive at a weighing station where a single operator will inspect, identify, and weigh the arriving meat cut. The automation of the weighing process on boning lines has traditionally been conducted on single-meat-cut production lines. However, due to spatial restrictions in many meat plants, there is a preference in the beef industry to operate multiple meat cut types simultaneously on a single processing line. This multi-meat-cut processing strategy has made the automation of meat cut identification extremely challenging as there is a high probability of incorrect meat cut identification; any proposed automated system must have a high level of accuracy in order to avoid misclassification and line downtime.

1.3.3 Labor Productivity

Labor productivity in the agricultural sector of Ireland has traditionally been lower than in the non-agricultural sector due to the nature of the production processes and inputs required. The labor productivity gap refers to the difference in labor productivity between two or more sectors of an economy. It typically arises due to differences in the nature of production processes, the level of technological advancement, and the amount and quality of capital and labor inputs used in each sector.

There has been a similar trend of a labor productivity gap between the agricultural and nonagricultural sectors in Ireland. According to data from the Central Statistics Office (CSO) of Ireland, labor productivity in the non-agricultural sector in Ireland has been consistently higher than in agriculture (Office 2023). This is largely due to the nature of the production processes and inputs required in each sector. The non-agricultural sector tends to be more capital-intensive and technology-driven, which can lead to higher labor productivity. In contrast, agriculture relies more on manual labor, which can be less productive. However, as mentioned earlier, labor productivity in agriculture has been increasing over time due to the adoption of new technologies and practices. Nonetheless, the labor productivity gap between the agricultural and non-agricultural sectors is still present in Ireland and has implications for income distribution and economic growth. It highlights the need for policies aimed at increasing labor productivity in the agricultural sector to ensure a more balanced and sustainable development of the economy.

1.4 Motivation

With the advent of digital photography in devices such as smartphones and digital security cameras, the volume of images produced on a daily basis has increased dramatically. The identification of objects within images and the classification of these objects has received a considerable amount of attention. For example, Deep Learning has been successfully used to classify images in areas such as autonomous vehicles or medical image classification (Guo, Shang, and Z. Li 2019; H. Wang et al. 2019). In the meat industry application of image-classification has predominantly been applied to product quality characteristics such as fat content or product blemishes but not to the specific identification of disparate products (Al-Sarayreh et al. 2018; Ropodi et al. 2015; X. Yu et al. 2018). However, in high-volume meat plants, the use of human operators to identify and weigh these products can increase the risk of cross-contamination between cuts and the reduction of human operators could have a beneficial impact on shelf life (Véronique 2008) and line efficiency.

The image data of topside beef cuts were collected to create a novel dataset containing five different product types, which will be described in Chapter 3. This provides a simulation of a real-world scenario where quality control staff identify different products simultaneously at the end of the product line, and is an ideal dataset to implement machine learning algorithms as part of the solution architecture for the automation of product identification.

1.5 Contribution

The main goal of this research is to automate the processes of identifying meat products. Removing a human operator will reduce the risk of cross-contamination across cuts and improve product shelf life along meat processing lines. In that context, the contribution can be articulated as follows:

- A novel ensemble machine learning classifier has been proposed that automates the identification of meat products using their images and weights, to deliver a high degree of accuracy.

- By comparing the performance of a number of machine learning methodologies, some new insights in terms of processing color and greyscale images of meat cuts have been uncovered.

- This research also generated a dataset containing 8,237 labeled images of five different beef products, which can be exploited by agricultural (abbr. Agri) researchers focusing on related topics.

- The evaluation had two objectives: to understand the effectiveness of different machine learning models for meat cut identification and thus demonstrate the efficacy of our method to automate product identification on beef boning lines.

Chapter 2

Literature Review

Machine Learning and, in particular, deep learning have been used to varying levels of success in determining food quality and food safety. In this section, how previous approaches used image processing within the food industry and in particular, their application in the meat industry are examined. This also sheds light on the techniques used for object identification and image detection also called image classification, imbalanced classification, ensemble methods, and meta-learners.

The following procedure was followed to choose related studies:

- **Defining a review protocol** states the research question being addressed and the methods that will be implemented to accomplish the review. In this research, the research question is "Can structured and unstructured data be integrated for imbalanced classification of meat-cut images?" The research method is experiment-based where an array of experiments has been carried out to answer the question.

- **Search strategy** aims to identify as much of the relevant literature as possible. In this research, databases such as Google Scholar, IEEE Xplore, and ResearchGate have been looked into to find relevant papers and articles. A number of keywords were used to choose the related study such as Quality OR Food Classification AND Machine Learning, Meat-Cut AND Image Identification OR Image Classification OR Object Detection, Image Classification AND

Imbalanced Dataset, Synthetic minority oversampling technique (SMOTE) OR Generative Adversarial Networks (GAN) AND Object Identification, etc.

- **Inclusion and exclusion** are the criteria that assess each potential primary study. To avoid the necessity for translation during the research, the scope of the search was restricted in this study to papers that were written in English. An effort was made to limit the literature from grey sources (such as moderated blogs) and peer-reviewed academic publications to those that offered high standards of quality in order to decide which sources were appropriate to employ throughout the proposed research.

2.1 Artificial Intelligence in Food and Meat Industries

The problem space for image detection within the food industry has been broadly broken into three criteria outlined by (L. Zhou et al. 2019), and they are as follows: product quality, food type classification, and product traceability.

Product quality has attracted a considerable volume of attention within the literature. For example, in (Al-Sarayreh et al. 2018), the authors implemented different machine learning techniques and a deep neural network on hyperspectral images (HSI) to detect adulteration in red meat. In (K. Chen et al. 2010), computer vision and machine learning methods were used for the color grading of beef fat. In (Y. Han et al. 2021), the HSI system along with deep learning techniques was implemented to estimate the quality of nuts. In (Ropodi et al. 2015), meat adulteration was identified in beef and pork products using samples from pure beef, pure pork, and adulterated meats from a multispectral imaging system by implementing machine learning tools such as partial least square discriminant analysis (PLS-DA) and linear discriminant analysis (LDA). Numerous other classifiers such as logistic regression (LR), decision trees (DT), k-nearest neighbor (KNN), artificial neural networks (ANN), and support vector machines (SVM), have been used in quality classification, identification of freshness, and nutrient prediction based on spectral data (L. Zhou et al. 2019). In (X. Yu et al. 2018), a regression-based classifier with stacked auto-encoders was implemented on the features extracted from HSI of shrimps to classify them as *fresh* or *stale*. The quality of meat can also be assessed by the level of fat content in addition to the composition of chemical indicators. The measurement and identification of the percentage of fat in pieces of meat were performed using dual-energy X-ray absorptiometry in conjunction with an array of machine learning algorithms (Sabol et al. 2006). These approaches all showed varying levels of success but in general, were either a binary classification problem or used machine learning solutions to measure the level of fat within a piece of meat and avoided the multi-class problem of product identification. Additionally, the use of HSI technology does not necessarily create an advantage over normal camera technology as the shape of the product will have more significance in identifying the product.

Food type classification frequently uses deep learning to determine if an image represents food or, alternatively, to identify the food type within an image. For example, the authors in (Jia et al. 2019) used transfer learning to identify the presence of food within images with a test accuracy score of 86.4% on the eButton dataset (Beltran et al. 2016) and 98.7% on the Food 5k dataset (Technology Lausanne n.d.). However, the training dataset for eButton data had a considerably higher accuracy score of 91.5%, which suggest the occurrence of overfitting. In (McAllister et al. 2018), the authors managed an improved result on the Food-5K dataset with an accuracy of 99.4% using a Radial Basis Function (RBF) kernel-based SVM with ResNet-152. As well as identifying food within an image, the ResNet and the DenseNet transfer learning algorithms were used to classify the Betawi traditional food image dataset into various food types (Setyono, Chahyati, and Fanany 2018). In this particular study, there was a wide variety of differing food types but the authors did not focus on the differences within food types, such as the different cuts of meat that can be created from a single primal cut in meat processing.

Product traceability has not been extensively researched using image detection techniques, although a small study by the researchers in (Hviid, Jørgensen, and Dahl 2011) and a subsequent larger study presented in (Larsen et al. 2014), were the first attempts to use computer vision techniques to track pork loins (40 and 211 loins respectively) along a slaughterhouse production line. This approach used DAISY image features (Tola, Lepetit, and Fua 2010) to generate Bag of Words descriptors of the meat cuts. However, this technology focused on production lines where only one outcome product was possible and did not take into account the multi-product nature of many meat plant boning lines.

AI has transformed the food and meat industries by leveraging machine learning, computer vision, and data analysis. It enables automated quality control, detecting contaminants, and spoilage, and ensuring food safety. AI-powered systems enhance production efficiency and optimize yield estimation and portion control. Computer vision algorithms enable accurate classification and sorting of meat products, reducing manual labor. AI also contributes to traceability, providing transparency in the supply chain and enabling quick identification of issues. Furthermore, AI-driven innovations improve consumer experience by personalizing recommendations and enhancing product development.

2.2 Image Identification

Image identification has been a topic of research for many years, with significant progress being made due to the advent of deep learning techniques. In this literature review, some of the recent developments in the field of image identification will be highlighted.

In recent years, Convolutional Neural Networks (CNNs) have become the dominant approach for image identification due to their ability to extract complex features from images. (Krizhevsky, Sutskever, and Hinton 2017) introduced the AlexNet CNN architecture, which achieved a significant improvement in image classification accuracy on the ImageNet dataset. Since then, many CNN architectures have been proposed that have achieved state-of-the-art performance on various image classification benchmarks. Some notable examples include VGGNet (Simonyan and Zisserman 2014), InceptionNet (Szegedy et al. 2015), and ResNet (K. He, X. Zhang, et al. 2016).

Recently, attention-based models have been proposed for image identification tasks. These models learn to attend to different parts of the image that are relevant to the task. One example is the Transformer model (Vaswani et al. 2017), which was originally proposed for natural language processing but has been adapted for image recognition by replacing the text input with image features.

Deep learning techniques have significantly advanced the state-of-the-art in object detection. The most successful approaches are based on Convolutional Neural Networks (CNNs) can be categorized into two groups: two-stage detectors and one-stage detectors.

Two-stage detectors first generate a set of region proposals and then classify each proposal as an object or background. The most popular two-stage detector is Faster R-CNN (Ren et al. 2015), which achieved state-of-the-art performance on the PASCAL VOC and MS COCO object detection benchmarks. Several extensions of Faster R-CNN have been proposed to further improve the accuracy and efficiency, such as Mask R-CNN (K. He, Gkioxari, et al. 2017) and Cascade R-CNN (Cai and Vasconcelos 2018).

One-stage detectors, on the other hand, directly predict the class and location of objects without the need for region proposals. The most successful one-stage detector is YOLO (Redmon, Divvala, et al. 2016), which achieved real-time performance on a single GPU. YOLO has several variants, including YOLOv2 (Redmon and Farhadi 2017) and YOLOv3 (Redmon and Farhadi 2018), which further improve the accuracy and speed of the original YOLO.

Recently, anchor-free object detectors have been proposed to further simplify the object detection pipeline by removing the need for anchor boxes. These detectors directly predict the location and size of objects without the need for anchor boxes. Some successful anchor-free detectors include CornerNet (Law and Deng 2018), CenterNet (X. Zhou, D. Wang, and Krähenbühl 2019), and FCOS (Tian et al. 2019).

Image identification is a rapidly evolving field with various techniques and architectures being developed and applied to different tasks. CNNs, transfer learning, attention mechanisms, and generative models are some of the popular techniques used in image identification, while interpretability and explainability continue to be active areas of research.

2.3 Imbalanced Classification

Imbalanced classification refers to a classification problem where the distribution of classes in the training dataset is not balanced, i.e., one class may have significantly more samples than the other(s). For example, in a medical diagnosis task, the majority of patients may not have a disease, while only a small fraction of patients may have the disease. This results in an imbalanced dataset where one class has a much higher frequency than the other.

Imbalanced classification can pose challenges for machine learning algorithms because they tend to be biased towards the majority class. This means that the model may perform well in identifying samples from the majority class but may perform poorly in identifying samples from the minority class. This is particularly problematic when the minority class represents the class of interest, such as detecting rare diseases.

The following methods are incorporated to address the imbalanced classification.

2.3.1 Dataset Geometry

In (Weiss 2004), the author discusses the concept of rarity within the context of the minority class. Rarity can be categorized into two types: absolute and relative. Absolute rarity refers to situations where the number of data points associated with a rare class or case is extremely small. Due to this scarcity of data, detecting patterns or regularities within the rare class becomes challenging. On the other hand, relative rarity occurs when the proportion of data points in one of the classes is significantly low, while the overall number of data points is high. It is important to note that imbalances between classes and within classes are interconnected. Therefore, addressing the issue of between-class imbalance automatically helps in managing the within-class imbalance as well.

The geometry of the minority class has been discussed in (Napierala and Stefanowski 2012). The authors identified that there are four ways in which minority classes can be categorized by analyzing a 2D visualization of a dataset trained by Multidimensional Scaling (MDS).

- **Safe** are the data points located in the homogeneous regions populated by the examples from a single class only.
- **Borderline** are the data points placed at the boundary of majority and minority classes.
- **Outliers** are the data points appearing deeper in the regions where the majority class prevails.
- **Rare** are some data points far from the clusters.

To identify the type of data points, an analysis of the class labels of their k-nearest neighbors was performed. For simplicity, the authors set the value of k to 5. It was also mentioned that a value smaller than 5 may poorly distinguish the nature of examples. A high value of k can lead to inconsistency with its local neighborhood. Table 2.1 shows the labels assigned to the data points as safe, borderline, rare, and outlier using the proportion.

(Lin et al. 2017) characterized the imbalanced classification as follows:

- *Class overlapping:* As shown in Figure 2.1, when the examples from different classes overlap, the learners have difficulties learning the attributes between

Table 2.1: Labels assigned to the data-proportion

Proportion	Labels
5:0 or 4:1	Labelled as safe (S)
3:2 or 2:3	Labelled as borderline (B)
1:4	Labelled as rare (R)
0:5	Labelled as outlier (O)

different classes. Predominantly, the examples belonging to the minority class are categorized into the majority class.

- *Small sample size:* Generally, during the data-mining process it is not always possible to collect sufficient data which is a major challenge as further data collection is not possible.

- *Small disjuncts:* In Figure 2.2, the minority class samples are distributed in numerous feature spaces which results in a high degree of compilation during the classification process.

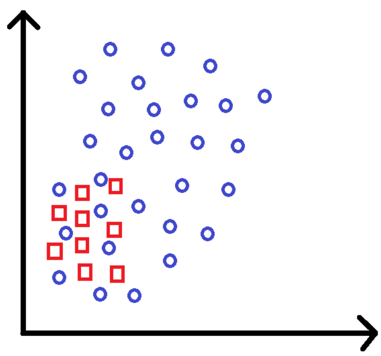


Figure 2.1: Class overlapping

In (Sampath et al. 2021) the imbalanced data has been classified as inter-class and intra-class imbalances. The inter-class imbalance lies with a binary class classification problem where one of the classes is having less number of data points. The

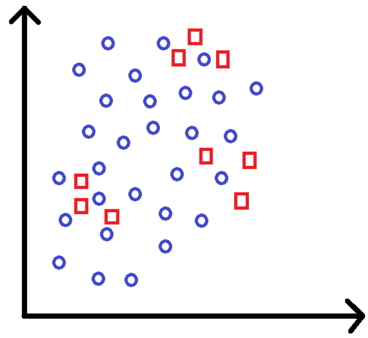


Figure 2.2: Class disjunct

imbalance can be described using proportion or imbalance ratio. When the imbalance lies within one of the classes, it is called an intra-class imbalance problem.

2.3.2 Data-level Approach

In the data-level approach, one aims to deal with the imbalanced nature at the beginning. Here the aim is to balance the data by reducing or increasing the number of data points. Re-sampling and synthesizing techniques are two extensively used methods.

Re-sampling In re-sampling, oversampling, under-sampling, and hybrid sampling are implemented. Resampling techniques aim to alleviate the effects of class imbalance by manipulating the dataset to create a more balanced representation of the classes. Resampling techniques aim to create a more balanced dataset by either increasing the number of samples in the minority class (oversampling) or reducing the number of samples in the majority class (undersampling). Oversampling techniques generate synthetic samples or duplicate existing minority class samples, while undersampling techniques remove instances from the majority class. By rebalancing the class distribution, resampling helps improve the performance and generalization of machine learning models, enabling them to learn from both classes effectively.

Oversampling: Over-sampling of imbalanced data is a technique used to address the problem of class imbalance by increasing the number of samples in the minority class. In situations where the minority class is underrepresented, over-sampling aims to create a more balanced dataset to mitigate the bias towards the majority class. By increasing the representation of the minority class, over-sampling enables the model to learn from a more diverse set of instances, improving its ability to recognize and classify minority class instances accurately. Random oversampling is one of the strategies that generate samples randomly using the bootstrap method (Menardi and Torelli 2014).

Under-sampling: Under-sampling of imbalanced data is a technique used to tackle class imbalance by reducing the number of samples in the majority class. When the majority class dominates the dataset, under-sampling aims to create a more balanced representation of the classes. This approach helps to address the bias that may arise due to the unequal distribution of classes. By reducing the dominance of the majority class, under-sampling allows the model to focus more on learning from the minority class, leading to better recognition and classification of minority class instances. Random under-sampling is an under-sampling strategy that removes the data points from the majority class randomly with or without replacement. For example, if there are 4000 majority examples and 500 minority examples, the dataset can be made a 50:50 ratio by removing 3500 examples from the majority examples (Mohammed, Rawashdeh, and Abdullah 2020; Shelke, Deshmukh, and Shandilya 2017; Lin et al. 2017; Guan et al. 2021). (Lin et al. 2017) implemented two strategies that employed a clustering algorithm to remove the samples from a majority class. In the first strategy, the k-means clustering algorithm was implemented where the number of clusters (i.e. k) was set to be the number of examples in the minority class (i.e. k=N). Then, the k cluster centroids are generated using the k-means algorithm over M data samples in the majority class. These centroids were used to replace the entire majority class examples which balanced the classes in an imbalanced dataset. The second strategy employed the method of Euclidean distance to calculate the similarity between cluster centroids and the data sample in the same cluster as it was found that the cluster centroids were the average of the data samples in a cluster and a new additional sample for the majority class. Similar data points were removed from the majority class.

The concept of Tomek link was used to remove the examples from the majority class while leaving the examples from the minority class untouched in (Tomek 1976). Tomek link, (C. Jiang et al. 2023), takes two points, x and$y$, each having different majority class then calculate distance, δ, between x and y. The pair (x, y) is called a Tomek link if no example z exists such that $\delta(x, z) < \delta(x, y)$ or $\delta(y, z) < \delta(y, x)$. Using Tomek links, the borderline examples and examples suffering from the class-label noise can be removed. To reduce the number of redundant samples, the creation of a consistent subset, C, from the training set, S, is proposed in which $C \subseteq S$ is consistent with S if, when used by the 1–NN rule, it correctly categorizes data points in S.

In condensed nearest neighbor (Susan and Kumar 2021), the nearest neighbors are removed based on the distance calculated, and the distance metric used is Euclidean. However, the steps for edited nearest neighbors are as follows:

1. For each i,
   (a) find the k-nearest neighbors to X_i among $\{X_1, X_2, ..., X_{i-1}, X_{i+1}, ..., X_N\}$
   (b) find the class θ associated with the largest number of points among the K-nearest neighbors, breaking ties randomly when they occur.
2. Edit the set $\{(X_i, \theta_i)\}$ by deleting (X_i, θ_i) whenever θ_i does not agree with the largest number of the K-nearest neighbors as determined in the foregoing.

Hybrid sampling: In this method, the oversampling and under-sampling are implemented as one followed by the other. There are various combinations, such as Random Oversampling with Random Under-sampling, Random Oversampling with Tomek Link, etc., that can be used to resample the training datasets. Initially, the data points belonging to the minority class are oversampled to reach a certain percentage of the total number of data points in the majority class. Subsequently, the majority class is down-sampled to align its number of data points with that of the minority class.

2.3.3 Synthesizing Techniques

The synthetic generation of data takes place in synthesizing methods. The synthetic minority oversampling technique (SMOTE) is a method that uses the nearest-neighbor concept to create data synthetically. (Chawla et al. 2002) proposed oversampling techniques, synthetic minority oversampling technique (SMOTE), nominal SMOTE (SMOTEN), and nominal and continuous SMOTE (SMOTE-NC). SMOTE is implemented to oversample the numerical attributes while SMOTEN oversamples only the categorical features. To oversample a mixture of both, SMOTE-NC is used. To oversample a categorical variable, the new examples are generated by picking the most frequent category of the nearest neighbors present. Equation 2.1 shows the mathematics of a synthetic example generation for continuous features.

$$x_{new} = x_i + y * (x_{zi} - x_i) \tag{2.1}$$

where, x_i is the minority sample, x_{zi} is the nearest data point, and y is a random number between 0 and 1. For example, in Figure 2.3, (Imblearn 2021), the green point between two minority data points, x_i and x_{zi}, is the new point generated which is labeled as x_{new}. If $x_i = (0.3, 3.5)$ and $x_{zi} = (0.4, 3.2)$, then the distance between the x-axis and y-axis are, $d_1 = (0.4 - 0.3) = 0.1$ and $d_2 = (3.2 - 3.5) = -0.3$, respectively. The generated point will be given as, $x_{new} = (0.3, 3.5) + [0, 1] * (0.1 - (-0.3))$.

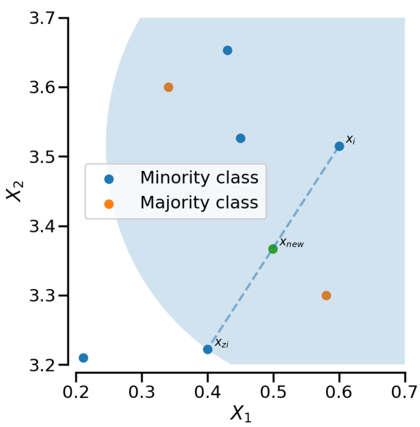


Figure 2.3: Synthetic Minority Oversampling Technique

Adaptive synthetic (ADASYN) was proposed in (H. He et al. 2008) that uses the k nearest neighbor rules and the SMOTE mathematics to generate new samples. It is an extension of SMOTE that addresses the challenge of imbalanced datasets with varying class densities. It focuses on generating synthetic samples in regions where the density of the minority class is low, giving more importance to these harder-to-learn instances.

(H. Han, W.-Y. Wang, and B.-H. Mao 2005) proposed borderline-SMOTE to oversample the minority examples on the borderline of the minority and majority classes. The method is based on finding the borderline minority examples and then selectively generating synthetic samples near the decision boundary to address the challenge of overlapping class distributions.

Data augmentation is a widely used technique in machine learning and computer vision to address the challenge of limited training data. It involves applying various transformations to existing data samples to create additional synthetic samples. By expanding the dataset with augmented samples, the model can learn more robust and generalizable representations. This literature review explores different data augmentation techniques and their impact on model performance.

One commonly used data augmentation technique is image augmentation, which involves applying transformations to images. One of the most common image augmentation techniques is random rotation, which involves rotating images to a certain degree. (Shorten and Khoshgoftaar 2019) demonstrated that applying random rotation improved the accuracy of deep learning models for object recognition tasks.

In addition to rotation and flipping, random cropping is a popular augmentation technique. (Han Zhang et al. 2019) proposed the Random Erasing technique, which randomly crops patches from images and replaces them with random noise. They found that Random Erasing improved the performance of object detection models.

Color jittering is another augmentation technique that alters the colorize of images. (Cubuk et al. 2018) introduced AutoAugment, an algorithm that automatically searches for the best color augmentation policies for a given dataset. They demonstrated that AutoAugment significantly improved the accuracy of image classification models.

Cutout is a technique where random patches of pixels are masked out from images. (DeVries and Taylor 2017) showed that Cutout regularized deep learning models, prevented overfitting and improved generalization performance. Moreover, there are domain-specific augmentation techniques. For medical image analysis, (Ronneberger, Fischer, and Brox 2015) proposed elastic deformation, which applies local deformations to medical images to simulate anatomical variations. They showed that elastic deformation improved the performance of deep learning models for medical image segmentation tasks.

Furthermore, generative adversarial networks (GANs) have been used for data augmentation. (Chamier et al. 2021) introduced CycleGAN, a GAN-based augmentation method that generates synthetic images from existing ones while preserving the original content. They demonstrated that CycleGAN effectively increased the diversity of training data and improved the performance of deep learning models for image classification.

Overall, image data augmentation techniques play a crucial role in deep learning for computer vision. The choice of augmentation techniques depends on the specific task and dataset. By effectively augmenting the training data, these techniques enhance model performance, improve generalization, and make deep learning models more robust.

Imbalanced classification methods address the challenges posed by imbalanced datasets where the distribution of class labels is skewed. These methods aim to improve the performance of machine learning models by effectively handling the minority class and reducing the bias towards the majority class. Various techniques are employed, such as resampling methods (oversampling or undersampling), cost-sensitive learning, ensemble methods, and algorithmic modifications. Resampling methods adjust the class distribution by either increasing the minority class samples or decreasing the majority class samples. Cost-sensitive learning assigns different costs to misclassification errors, giving more importance to the minority class. Ensemble methods combine multiple classifiers to leverage their complementary strengths. Algorithmic modifications adapt existing algorithms to handle imbalanced data, such as modifying the decision thresholds or introducing class weights. These methods enhance model performance, promote better generalization, and mitigate the impact of class imbalance in classification tasks.

2.4 Ensemble Methods

Ensemble methods are widely used in machine learning to combine multiple models and improve the overall predictive performance. In this literature review, some of the recent developments in the field of ensemble methods will be discussed.

Bagging and Boosting are two of the most popular ensemble methods. Bagging involves training multiple models independently and combining their predictions by averaging or voting. Random Forest (Breiman 2001) is a popular bagging algorithm that builds decision trees on random subsets of the data and aggregates their predictions. Boosting, on the other hand, involves iteratively training models that focus on hard-to-predict examples. The most successful boosting algorithm is AdaBoost (Freund and Schapire 1997), which assigns higher weights to misclassified examples and trains subsequent models on the re-weighted data.

Another popular ensemble method is Stacking, which involves training multiple models and using their predictions as inputs to a higher-level model. Stacking has shown promising results in several machine learning competitions, such as the Netflix Prize and the KDD Cup.

Recently, neural network-based ensemble methods have gained attention due to their superior performance in many tasks. One of the most successful neural network-based ensemble methods is the Deep Ensemble (Lakshminarayanan, Pritzel, and Blundell 2017), which trains multiple neural networks with different architectures and combines their predictions using Bayesian model averaging. Another neural network-based ensemble method is the Snapshot Ensemble (Huang et al. 2017), which trains a single network with multiple learning rates and snapshots its weights at regular intervals.

Ensemble methods have been widely used in machine learning to improve predictive performance. Bagging and Boosting are two popular ensemble methods, while Stacking and neural network-based ensemble methods have shown promising results in recent years. Further research in the area is likely to lead to even more effective ensemble methods.

2.5 Meta Learners

Learning to learn is called meta-learning where a new task is learned using the systematic observations from meta-data of previously learned algorithms. It is a subfield of machine learning (ML) that aims to improve the efficiency and generalization of learning algorithms by enabling them to learn how to learn from past experiences. Meta-learning has been an active area of research in ML and AI for several decades. In this literature review, the recent advancements and key concepts in meta-learning will be reviewed. Meta-learning can be seen as a higher-order learning process, where the goal is to learn how to learn new tasks efficiently and effectively. In traditional machine learning, the goal is to learn a fixed model that performs well on a specific task, given a set of training examples. However, in the real world, one encounters a large variety of tasks that require different types of models and strategies. Therefore, meta-learning has emerged as a promising approach to address the challenge of learning across different tasks and domains.

Meta-learning algorithms can be categorized into three main types: model-based, metric-based, and optimization-based. Model-based approaches use Bayesian inference to learn a distribution over models or model parameters. Metric-based approaches learn a distance metric between examples or tasks, which can be used to generalize to new tasks. Optimization-based approaches learn a set of hyperparameters or optimization algorithms that can be applied to different tasks.

Meata-learning approaches make learning much faster compared to machine learning algorithms and improve the design of ML pipelines. Meta-learning is a data-driven approach that replaces hand-engineered algorithms with novel approaches (Vanschoren 2018). The difference between a base-learning and a meta-learning is the scope of the level of adaptation: meta-learning focuses on choosing the right bias dynamically while base-learning fixes the bias a priori or user parametrized (Vilalta and Drissi 2002).

2.5.1 Meta-Features

In a meta-learning framework, the initial task is to collect meta-data that explains previously learned models on various tasks. These include the same configuration of algorithms used for training models such as setting hyper-parameters, composing pipelines, the architecture of networks, resulting model evaluations like time to train and accuracy, and learned parameters like trained weights are called *meta-features*.

2.5.2 Meta-Learners with Imbalanced Classification

(Z. Liu et al. 2020) created a meta-learner for imbalanced classification tasks with three parts, meta-sampling and ensemble training to build ensemble classifiers and meta-training to optimize the meta-sampler. The meta-state provided information about the bias/variance of the ensemble models. This research focused only on the binary classification tasks and did not perform any multi-class classification.

Meta-learners are powerful techniques that aim to improve the performance and generalization of machine learning models by learning how to learn. These algorithms operate at a higher level by leveraging insights from multiple base learners and combining their predictions or models to make more accurate and robust decisions. Meta-learners can adapt to changing data distributions, handle limited labeled data, and enhance model transferability across different tasks or domains. By effectively utilizing the knowledge acquired from base learners, meta-learners provide a means to optimize model architectures, hyperparameters, or feature representations, ultimately leading to improved model performance and more efficient learning processes.

Table 2.2 shows the difference between Meta learners and Ensemble methods.

Table 2.2: Meta Learners vs Ensemble Methods

Meta Learners	**Ensemble Methods**
Meta-learners are algorithms or frameworks designed to learn from multiple learning tasks in order to improve the learning process or adapt to new tasks more efficiently. They aim to learn higher-level knowledge or representations that can be applied across different tasks.	Ensemble methods, on the other hand, involve combining multiple models (learners) to create a more powerful or accurate prediction. The goal of ensemble methods is to leverage the diversity of individual models to enhance overall performance.
Meta-learners typically focus on learning how to learn or adapt. They often employ techniques such as gradient-based optimization, memory-augmented architectures, or reinforcement learning to optimize the learning process. Meta-learners can be trained on a set of tasks and then generalize that knowledge to new, unseen tasks.	Ensemble methods combine the predictions of multiple models, known as base learners or weak learners, to generate a final prediction. Each base learner is trained independently on the same or different data subsets, and their predictions are aggregated using various techniques such as majority voting, weighted averaging, or stacking.
The primary objective of meta-learners is to improve generalization and adaptability to new tasks. They aim to capture patterns or relationships across multiple tasks and leverage this knowledge to quickly adapt to new, unseen tasks with limited training data.	Ensemble methods primarily aim to enhance predictive accuracy and reduce overfitting. By combining multiple models, ensemble methods can reduce the impact of individual model biases and errors, leading to improved overall performance.
Meta-learners are often applied in few-shot learning, transfer learning, and reinforcement learning domains. They excel in scenarios where training data is limited or tasks exhibit certain similarities.	Ensemble methods have a wide range of applications and are commonly used in various machine learning tasks such as classification, regression, and anomaly detection. They can be applied to any problem where multiple models can be trained and combined.

2.6 Summary

Ensemble methods, transfer learning, and large-scale deep neural networks have been deployed in numerous image detection problems both within and outside the meat sector, but none of the approaches studied have applied these methods to identify individual boneless beef cuts and in many cases, the volume of data used was quite small. In contrast, the research presented in this thesis assesses the application of deep learning in detecting products from images that in many cases are highly homogeneous.

Chapter 3

Dataset

The first stage is to gather a sizable and varied dataset of meat cut images. The collection of meat-cut image data is crucial for experimental studies on the meat business, food science, and animal husbandry. It entails taking pictures of different slices of meat for research and development purposes.

The primary benefit of collecting meat-cut image data is the visual documentation of various meat cuts and their features. The shape, size, marbling, fat distribution, and texture of meat pieces might vary. Researchers can build a comprehensive dataset that appropriately portrays the wide variety of meat products available by gathering high-quality photos of various meat slices. These images act as a visual reference for researchers, but they also serve as a valuable resource for the industry, aiding in quality control, standardization, and meat classification.

Accurate and standardized meat-cut images are vital for quality control and product consistency in the meat industry. Meat processors and retailers rely on consistent visual standards to classify and grade meat products. By collecting and analyzing meat-cut images, researchers can develop objective criteria for categorizing meat cuts based on visual attributes. This enables the industry to maintain consistent quality standards, improve consumer trust, and ensure that consumers receive the expected meat product based on visual appearance.

The collection of such data is also essential for the study of meat and food technologies. Researchers investigate a number of variables, such as animal breed, age, food, and post-mortem procedures, that affect the quality of meat. Researchers can examine the visual characteristics and correlate them with the associated meat quality indicators by gathering photos of meat cuts from various sources and under various circumstances. This enables a greater comprehension of the connection between meat quality and its visual appeal, resulting in advancements in meat-processing methods, product creation, and customer pleasure.

Furthermore, this is valuable in consumer studies and sensory analysis. Visual appeal plays a significant role in consumers' perception and acceptance of meat products. By presenting consumers with meat-cut images and collecting their feedback, researchers can study the influence of visual cues on consumer preferences, purchasing decisions, and sensory perception. This information helps guide marketing strategies, product design, and consumer-oriented innovations in the meat industry.

Meat-cut image collection also supports the development of automated meat grading and sorting systems. With advancements in computer vision and machine learning, researchers can train algorithms to recognize and classify meat cuts based on their visual features. The availability of a comprehensive meat-cut image dataset enables the training and evaluation of such algorithms, leading to the development of efficient and accurate automated grading and sorting systems. These systems offer benefits such as increased productivity, reduced labor costs, and improved consistency in meat processing operations.

Additionally, it facilitates education and training in the meat industry. Students, researchers, and professionals in meat science and food technology programs require access to authentic and diverse meat-cut images for educational purposes. A well-curated dataset of meat-cut images allows for hands-on learning, fostering a better understanding of meat anatomy, identification of different cuts, and the relationship between visual appearance and quality characteristics.

3.1 PaceLine Debonning

PaceLine deboning is a method used in the beef industry that utilizes a conveyor belt system to transport carcass portions through the cutting room. The conveyor belt serves a dual purpose, not only transporting the product but also serving as the work surface for the deboners.

As shown in Figure 3.1, the speed of the conveyor belt determines the pace at which the work is performed on the deboning line. Each employee in the PaceLine has a designated task in the deboning process, and they must complete their task within the timeframe of the specific piece of meat passing by them on the conveyor belt. This synchronized workflow ensures that the deboning process is efficient and consistent.

One of the advantages of the PaceLine deboning method is that it allows for efficient and uniform deboning with a potentially lower-skilled workforce. The individual tasks in the process are specific and straightforward, making them easier to learn and perform. This enables companies to train personnel quickly, reducing the need for highly skilled and experienced deboners.

Once the meat is deboned, it is typically fed into crates or onto a conveyor belt for further processing, storage, or packing. This allows for the seamless integration of the deboned meat into subsequent stages of the production process.

Overall, PaceLine deboning offers several benefits to the beef industry. The use of a conveyor belt system ensures a consistent pace of work, optimizing productivity. The simplicity of the individual tasks makes it easier to train personnel and maintain a skilled workforce. Additionally, the efficient deboning process and subsequent handling of the meat facilitate its storage, packing, and further processing.

3.2 Data Collection

Deductive Analytics for Tomorrows Agri Sector (DATAS) (McCarren, S. McCarthy, et al. 2017) is a project sponsored by the government of Ireland to create Agricul-

Figure 3.1: PaceLine Deboning

tural/Food related datasets that can be used to enhance practitioners' insights in the Agri sector. The data collected for this project range from international macro economic indicators to micro data collected at the plant level. One such dataset was the collection of data from beef cuts taken from the Topside trimming line of a major Irish beef processor. In most Irish meat plants, a *Topside* cut is an inner muscle taken from the hind quarter of a bovine animal. The process flow for this line required an operator to weigh the primal topside cut on a Start of Line (SOL) weighing scale. Each cut was then placed on a conveyor belt where a team of operators removed fat, gristle, or secondary muscles. The remaining primary muscles were then labeled and weighed, and an image was captured by a trained operator at the End of Line (EOL) weighing scales. For this particular study, there were five different types of the cut as shown in Figure 3.5: (a) is a Cap Off Pear Off, PAD topside muscle (product 20001); (b) is a Cap off, Pear on topside muscle (20002); (c) is a Topside Heart muscle (20003); (d) is a topside Bullet muscle (20004); and (e) is a Cap Off, Non-Pad, Blue Skin Only topside muscle (20010).

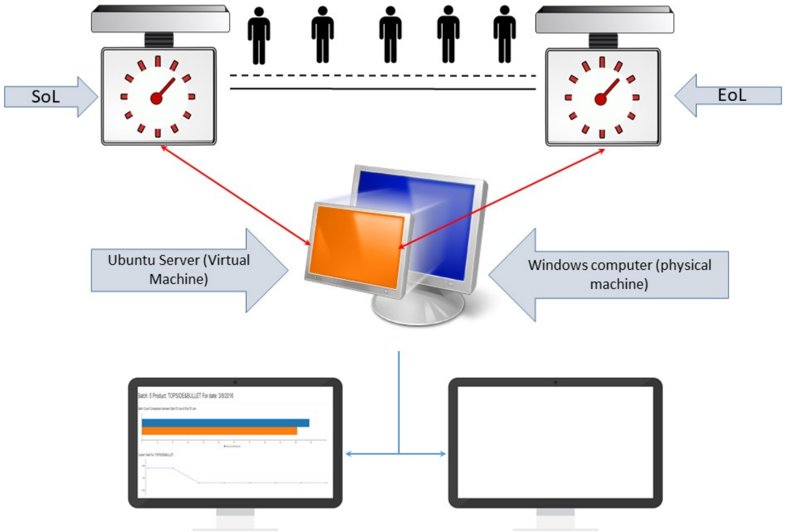


Figure 3.2: Data Collection Process

For this study, data acquisition required a hardware setup of weighing scales (System 1985) at both the Start of Line (SOL) and End of Line (EOL) together with a Vivotek bullet camera (Vivotek n.d.) at the EOL to capture a photo image of each meat cut as shown in Figure 3.2.

3.2.1 Weight Collection Using DEM Weighing Scale

DEM weighing scale, also known as dynamic electronic measuring weighing scale, is a type of scale that is used to measure the weight of objects or materials in motion. It is commonly used in industries such as logistics, manufacturing, and transportation, where weighing objects while they are in motion is necessary. An overview of how a DEM weighing scale typically works is as follows:

- Load Cell: The DEM weighing scale consists of load cells, which are sensors that convert the force or weight applied to them into an electrical signal. Load cells are strategically placed at specific points on the scale or conveyor system to accurately measure the weight of the moving objects.

- Conveyor Belt or System: The objects to be weighed are typically placed on a conveyor belt or passed through a conveyor system. As the objects move along the conveyor, they come into contact with the load cells.

- Measurement and Data Processing: When an object comes in contact with a load cell, it applies force to the load cell, which then generates an electrical signal proportional to the weight of the object. These electrical signals are collected and processed by the weighing scale's electronics.

- Calibration and Adjustment: To ensure accurate measurements, DEM weighing scales require calibration. Calibration involves setting the scale to zero when there is no load on the conveyor system and verifying that the scale provides accurate measurements at specific load points.

- Display and Data Output: The measured weight data is displayed on a digital or graphical interface. This display can show the weight in various units, such as kilograms or pounds. Additionally, DEM weighing scales often have data output capabilities, allowing the weight data to be sent to other systems or devices for further analysis or integration into larger operational processes.

- Control and Feedback: DEM weighing scales can be integrated into control systems to automate processes based on weight measurements. For example, in a manufacturing environment, the weight data from the scale can trigger actions such as sorting, packaging, or quality control checks.

The weight of each product was collected using DEM weighing scale. Table 3.1 shows a description of these. In Figure 3.3, the box plot shows that there are no outliers in the collected weights.

3.2.2 User Interface for Data Collection

In addition, bespoke data capture software using a node.js platform (Cantelon et al. 2013) was used to acquire the characteristics of each product being weighed.

1. The first step sees a manual capture of: *weight*, *batch number* and *time* at the SOL scales.

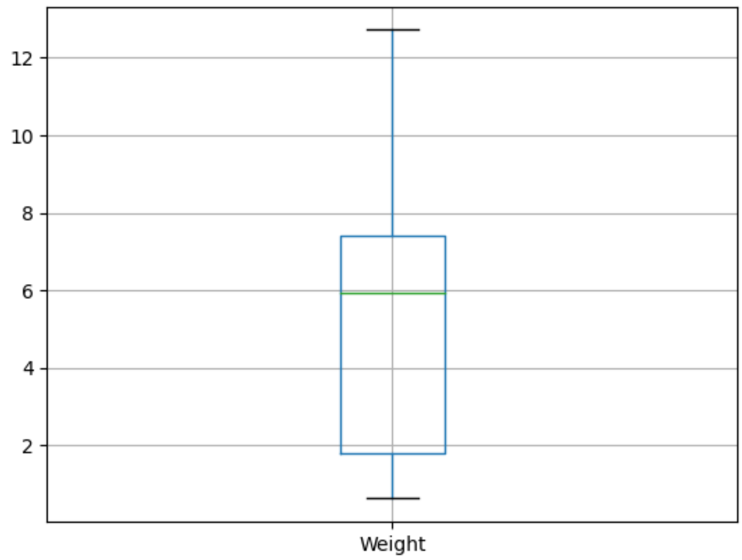


Figure 3.3: Box Plot of Weights

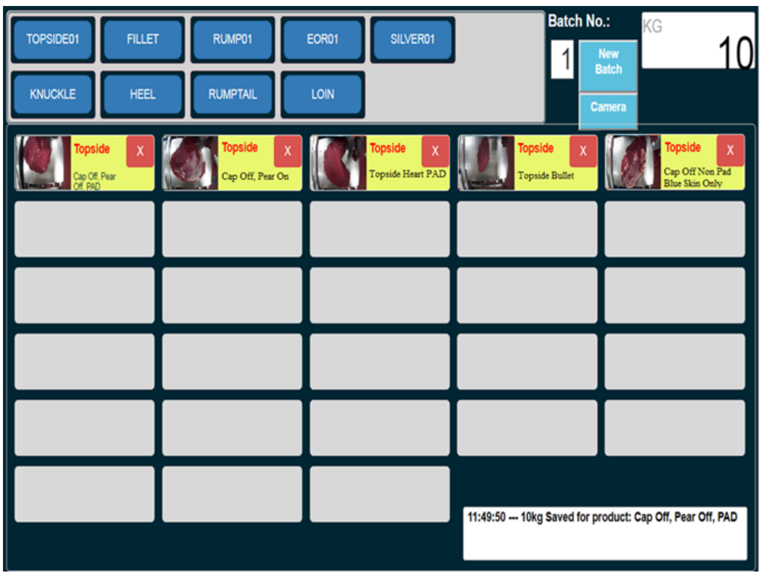


Figure 3.4: EOL User Interface for Data Collection

Table 3.1: Descriptive Statistics of Weights Collected

Metric	**Value**
Mean	5.36
Std Dev	2.80
Min	0.62
25%	1.80
Median	5.92
75%	7.39
Max	12.72

2. The next step sees a software-based capture of: *time*, *operator*, *batch number*, *product label*, *weight* and *photo image* at the EOL scales.

3. As the variables are captured using different devices at step 2, both SOL and EOL variables must be linked to the image capturing at Step 3. The EOL operator identifies the product using the data capture interface (shown in Figure 3.4), ensuring the correct image is stored to disk and linked to the appropriate database entry containing the variables captured at both EOL and SOL points.

4. After each product is removed from the scales, an image of the empty scales is captured for usage in background removal during the pre-processing step discussed in section 3.4.

The user interface for the data capture software can be seen in Figure 3.4. A trained operator identifies the products and selects one of the buttons from *Cap Off, Pear Off, PAD*, *Cap Off, Pear On*, *Topside Heart PAD*, *Topside Bullet* and *Cap Off Non Pad Blue Skin Only*, to align the images and variables described above. The data collection period lasted 3 weeks and the equivalent weights were collected for each cut from the topside trimming line over this period. A summary of the data captured is shown in table 3.2, where N is the frequency distribution of images, and $\bar{X}$ and S are mean and standard deviation of weights respectively.

Table 3.2: Dataset Analysis

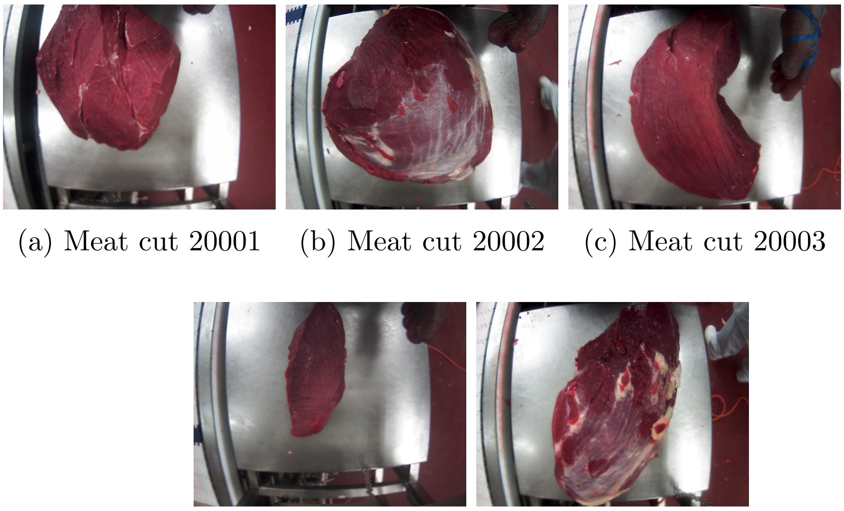
Product ID	N	**Product Description**	$\bar{X} \pm S$ **(kgs)**	**Cut Yield(%)**
20001	1060	Cap Off, Pear Off, PAD	6.47 ± 1.17	55.11
20002	14	Cap Off, Pear On	8.87 ± 0.98	68.18
20003	2132	Topside Heart PAD	5.78 ± 1.10	44
20004	2085	Topside Bullet	1.40 ± 0.29	9.45
20010	2696	Cap Off Non Pad Blue Skin Only	7.82 ± 1.59	61.55

At the end of the data collection period, a yield analysis was conducted to determine if there were any outlying weights by comparing the weights on the SOL scales with the weights of the product on the EOL scales. As the beef plant operator had a specification limit of 10% for each of the products used in these experiments, any absolute difference between the yield value and the expected yield that exceeded 10% was flagged as an outlier and subsequently, removed from the dataset. As a result, 7,987 records were deemed acceptable for the final dataset (McCarren, Scriney, et al. 2021).

3.3 Dataset Features

In general, image datasets contain various features that are essential for training and evaluating computer vision models. These features provide important information about the images and enable the models to learn and make accurate predictions. The data collected in this research has

- **Classes/Categories:** Image datasets are typically organized into different classes or categories. Each image is assigned a specific class label indicating the object, scene, or concept it represents. This data has five mutually exclusive classes with their product ID label.



(a) Meat cut 20001 (b) Meat cut 20002 (c) Meat cut 20003

(d) Meat cut 20004 (e) Meat cut 20005

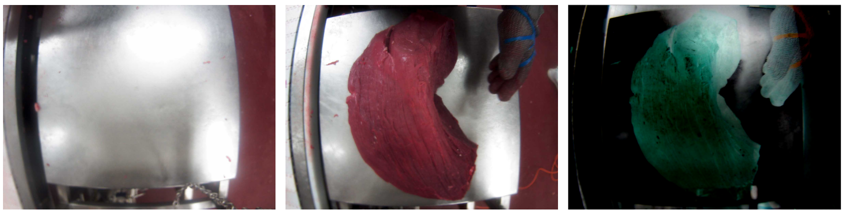
Figure 3.5: Topside Cuts: 5 meat cut variations.

- **Metadata:** Image datasets may contain additional metadata associated with each image, providing supplementary information about the images. In this dataset, the background images have also been collected.

3.4 Image Pre-processing

This section begins with an outline of the pre-processing steps required for image data before describing the implementation of the three separate deep learning models and a fourth approach, an ensemble method, used to detect the meat cuts within the image data.

When conducting image pre-processing, one generally aims to improve the prediction process by enhancing certain characteristics and/or blurring others (Lancaster et al. 2018). For this study, each meat cut image was accompanied by its associated background image such as that shown in Figure 3.6a. In order to remove distracting or confusing items such as operator hands or small meat blobs, the background image (3.6a), was removed from the image meat cut in 3.6b, resulting in Figure 3.6c. This image was then converted to greyscale as shown in Figure 3.6d, and finally, the edges were enhanced as shown in Figure 3.6e using Gaussian blur and edge detection techniques.



(a) Background (b) Scale with cut (c) Difference

(d) Greyscale image (e) Detecting Edges

Figure 3.6: Images At Various Stages of Pre-processing

Theoretically, with image identification algorithms such as convolutional neural networks, there is no need to engineer features during this process, as the mix of the convolution kernels and max pooling automatically creates features that can be inserted into a typical neural network (H. Liu et al. 2019). However, neural networks are highly non-linear, and estimating the choice of initial weights can be computationally expensive. Creating a simplified set of initial features, such as the object extremities, and using these as inputs to a basket of simpler algorithms or an ensemble of algorithms has been found to be successful in other applications (R. Wang, W. Li, and L. Zhang 2019). In order to identify these object extremities, images were standardized by rotating them so that the longest side was always in a vertical position. Figure 3.7 shows the co-ordinates and the virtual box drawn around the meat cut. From this image, the following features were calculated:

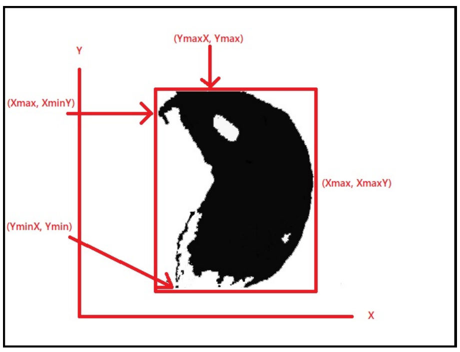


Figure 3.7: The co-ordinates and the virtual box around the meat cut

- **Density**: white pixel count relative to the total number of pixels.
- (X_{min}, $X_{min}Y$): the minimum X and the corresponding Y coordinate.
- (X_{max}, $X_{max}Y$): the maximum X and the corresponding Y coordinate.
- ($Y_{min}X$,Y_{min}): the minimum Y and the corresponding X coordinate.
- ($Y_{max}X$, Y_{max}): the maximum Y and the corresponding X coordinate.

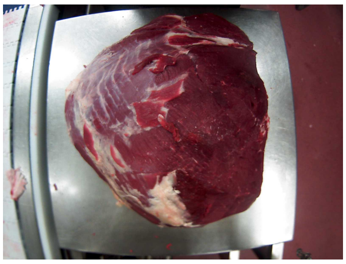
3.5 Handling Imbalanced Class

The frequency breakdown for the products shown in table 3.2 highlights product 20002 as having a disproportionately lower volume of occurrences. As the distribution of product 20002 in the classification problem is heavily skewed, this is called as an imbalanced class. This class imbalance can pose challenges for machine learning algorithms, as they tend to have a bias towards the majority class and may struggle to accurately predict the minority class.

It was decided to use data augmentation to create artificial training samples for 20002 in order to improve the imbalanced nature of the dataset. As part of the augmentation process, transformations such as anticlockwise rotation, clockwise rotation, horizontal flip, vertical flip, noise addition, and blurring were implemented. These processes created 84 additional images for product 20002 providing a final count of 98 images for product 20002.

3.6 Summary

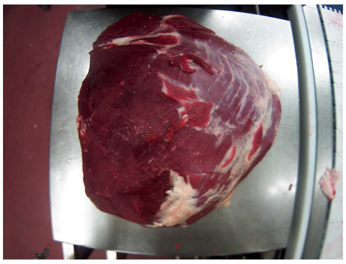
The collection of meat image data is crucial for the development of artificial intelligence models and systems in the domain of food processing, quality control, and agriculture. The data can be described as "Big," indicating a large number of images summing up to 7,987, which provides a substantial dataset for training and analysis. Furthermore, the data can be characterized as "Diverse," representing samples



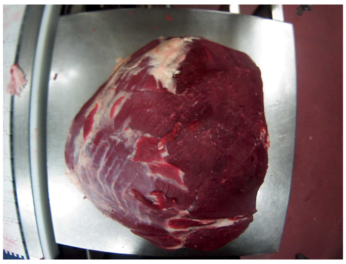
(a) Original Product

(b) Anticlockwise rotation

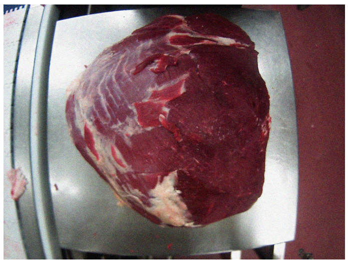
(c) Clockwise rotation



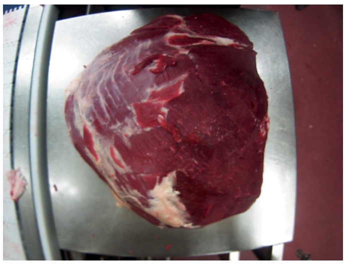
(d) Horizontal flip



(e) Vertical flip



(f) Noise addition



(g) Blurring

Figure 3.8: Image Augmentation

from five distinct categories, likely enhancing the representativeness and variability of the dataset. The mentioned attributes of the dataset can contribute to various applications such as automated inspection and sorting, yield estimation and portion control, traceability and food safety, as well as research and innovation in the field.

Chapter 4

Methodology

In this section, the effectiveness of several machine-learning algorithms for identifying beef cuts, as described in Chapter 3, is evaluated. Taking into account the insights shared in Chapter 2, convolutional neural networks (CNNs) will be applied to process the structured image data. Additionally, a transfer learning technique known as ResNet (short for residual network) will be incorporated for its efficient training capabilities. The weights obtained from ResNet will be combined with the CNN model, and subsequently, a novel ensemble method, that leverages the predictions from the algorithms and their corresponding weights, will be introduced.

4.1 Convolutional Neural Networks

The Neural Network family of machine learning algorithms has evolved from the original Artificial Neural Network (ANN) to Deep Learning Architectures such as the Convolutional Neural Network (CNN) (Shah and Gandhi 2004; Hang et al. 2019). The CNN algorithm has shown particular success in identifying objects *within* images (Wallelign, Polceanu, and Buche 2018). The CNN algorithm processes data by passing images through multiple convolutional and pooling layers and applies non-linear transformations such as the Softmax or ReLU function to obtain the probability-based classes (X. He and Y. Chen 2019; Prabhu 2018). The elements of CNN are as follows:

- Convolutional Layers: CNNs utilize convolutional layers as their fundamental building blocks. The core operation of a convolutional layer is the convolution operation. It involves applying a set of learnable filters (also known as kernels or feature detectors) to the input data. This operation enables the network to learn hierarchical representations of visual features with increasing complexity.

- Pooling Layers: Pooling layers are often inserted between convolutional layers to downsample the feature maps, reducing their spatial dimensions. Max pooling and average pooling are common techniques used in CNNs to retain the most salient features while reducing computational complexity.

- Activation Functions: Non-linear activation functions, such as ReLU (Rectified Linear Unit), are typically applied element-wise after each convolutional or fully connected layer. ReLU introduces non-linearity and helps the network model more complex relationships between features.

- Fully Connected Layers: Towards the end of the network, one or more fully connected layers are commonly employed to aggregate the spatial information and make predictions based on the learned features. These fully connected layers are often followed by a softmax activation function for multi-class classification.

- Loss Function and Optimization: CNNs are trained using supervised learning, where a loss function, such as cross-entropy, is used to measure the difference between predicted and true labels. The network parameters are updated iteratively using optimization algorithms like stochastic gradient descent (SGD) or its variants to minimize the loss.

The functional form of a convolution layer is described in equation 4.1.

$$X_j^l = g\left(\sum_{i \in N_j} X_i^{l-1} * W_{ij}^l + B_j^l\right) \tag{4.1}$$

In equation 4.1, X_j^l is an output vector, W_{ij}^l is the convolution kernel, also known as weights, X_i^{l-1} is the previous layer's feature map, B_j^l is an additive bias given to each output map, N_j represents the selection of the input maps, and $*$ represents the convolution operation.

In a neural network, regularisation is a technique to prevent overfitting. Overfitting occurs when the model is over-parameterized relative to the volume of data available. Typically one finds that the training loss continues to reduce as validation loss remains the same and the final model selection does not perform well on the test data. A solution that is regularly proposed to prevent overfitting is the addition of dropouts within the input and flatten layers (N. Srivastava et al. 2014). A loss function describes the deviation of predictions from the ground truth (Zhao et al. 2016), and is required to calculate the model error. The error for a single pattern can be expressed as in equation 4.2, where λ is a user-defined parameter that controls the trade-off and α_i is the weights for a given output.

$$\widetilde{\epsilon}^n = \epsilon^n + \lambda \sum_{i,j} |(\alpha)_{ij}| \tag{4.2}$$

After each step, the parameters and learning rates get updated in order to minimize the error using algorithms such as Adaptive Moment (Adam), which is a first-order gradient-based optimization of the stochastic function and is based on adaptive estimates of lower-order moments (Kingma and Ba 2015). ReLU, a computationally inexpensive activation function, accelerates the training procedure by avoiding the vanishing gradient problem (X. He and Y. Chen 2019). In order to avoid overfitting, a CNN architecture, which was originally used to identify numbers in the MNIST dataset (Garg et al. 2019) by adding max-pooling and a dropout on each convolution layer (Park and Kwak 2016), was adapted.

4.2 CNN mixed with Product Weights

To create a model that incorporates both unstructured data i.e., images, and structured data i.e., weights, first a CNN branch was defined that processes the image data, utilizing convolutional layers, pooling layers, and fully connected layers. Simultaneously, a separate branch was constructed to handle the CSV data, which may consist of fully connected layers or other appropriate layers for structured data. The outputs from both branches are concatenated by merging the learned representations from both the image and weight data. To further enhance the model's capabilities, additional layers such as fully connected layers or softmax layers can be added after the merging point to facilitate predictions or perform the desired task.

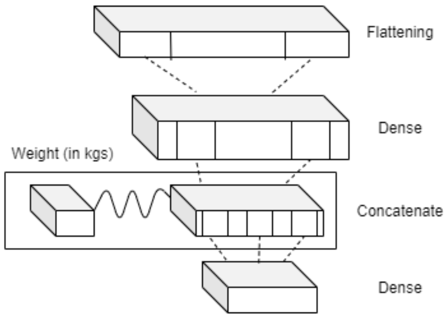


Figure 4.1: Architecture with Product Weight Infusion

Once the model architecture is defined and the branches are combined using the Parallel API, the next step is to compile the model. This involves specifying the loss function, optimizer, and evaluation metrics that will be used to train and evaluate the model. The loss function determines how the model's predictions are compared to the true labels, while the optimizer controls how the model's parameters are updated during training. Additionally, evaluation metrics provide insights into the model's performance during the training and evaluation phases.

With the model compiled, the training process begins. The combined image and CSV data are fed into the model, ensuring that the multiple inputs and outputs are appropriately handled. The model learns from the training data by iteratively adjusting its parameters based on the defined loss function and optimizer. This process continues for a certain number of epochs or until a convergence criterion is met.

After training, it is crucial to evaluate the model's performance on unseen data. This evaluation is typically done using a separate validation dataset. The model's predictions are compared against the true labels, and various evaluation metrics are calculated to assess its accuracy, precision, recall, or other relevant performance measures. Based on the evaluation results, the model may require fine-tuning to improve its performance. This can involve adjusting hyperparameters, such as learning rate or regularization strength, or modifying the model architecture itself to optimize its performance on the target task.

By following this approach of compiling the model, training it with combined data, and evaluating its performance, practitioners can iteratively refine and enhance the model's effectiveness. Fine-tuning based on evaluation results helps ensure that the model generalizes well and achieves optimal performance on the desired task.

As explained in the previous section, the weights (kgs) and images of the meat cuts were recorded simultaneously. In the meat industry, cuts are generally extracted from *primal cuts*, and knowing the weights of these cuts can potentially help in the identification of candidate labels. Integrating external models or input nodes such as the weights into the hidden layers of a Deep Learning architecture has been shown to be successful in previous research (Shi et al. 2020).

In figure 4.1, the weight of each meat cut has been integrated into the flatten layer of CNN mentioned in section 4.1. Flattening the final convolution layer converts the images into a 1-dimensional array and transfers it to the fully connected, dense layer. The weight is concatenated with 1-dimensional features and the last dense layer is used as an output layer which predicts the classes of the meat cut images.

4.3 Ensemble Approach with Product Weights

The ensemble architecture is presented in figure 4.2 as a 5-layer structure. At the first layer (Training: Data-Level 1), the handcrafted features x1, x2, ..., x8, were used in conjunction with each product weight, together with a basket of machine learning approaches to identify each meat cut. The three base learners shown at layer 2, were Multinomial Logistic Regression (MLR), Decision Tree Classifier (DTC), and CNN.

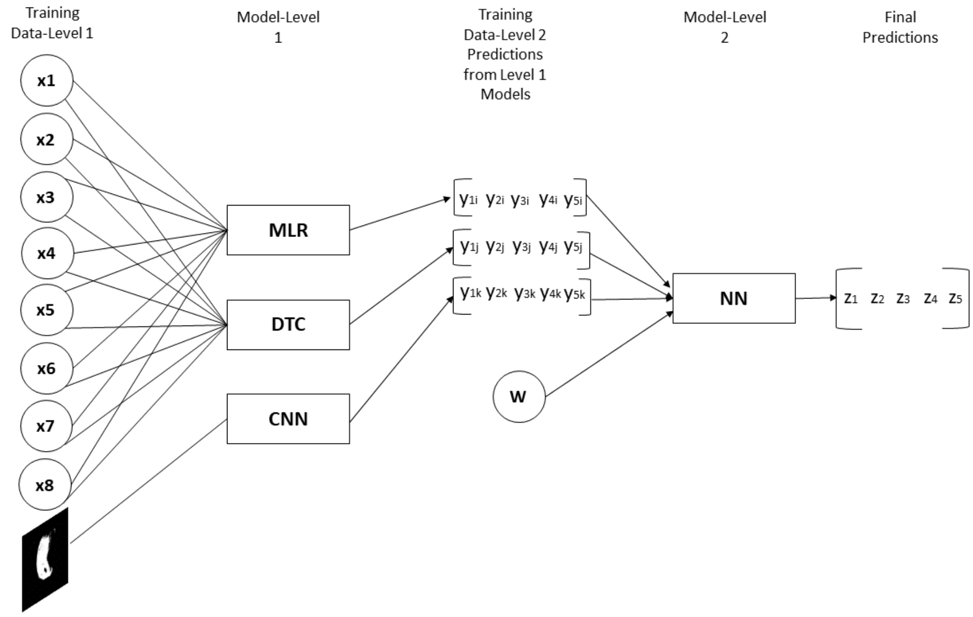


Figure 4.2: Ensemble Architecture using Multinomial Logistic Regression, Decision Tree classifier and CNN Learners

4.3.1 Base Learners

In machine learning, a base learner refers to an individual model or algorithm that serves as a building block within an ensemble learning framework. It is the basic or elementary model that forms the foundation for more complex learning architectures.

The base learner is typically a relatively simple and weak model that may not perform optimally on its own. However, when combined with other base learners or integrated into an ensemble, it contributes to the overall predictive power and generalization ability of the ensemble.

The choice of base learner depends on the specific problem domain and the characteristics of the data. Common examples of base learners include decision trees, logistic regression models, support vector machines, or neural networks with a small number of layers. These models are often selected for their simplicity, speed, or interpretability.

Ensemble methods, such as bagging, boosting, or stacking, are commonly used to combine the outputs of multiple base learners. The diversity and complementary strengths of different base learners can help mitigate individual model weaknesses and improve the overall predictive performance of the ensemble.

Base learners play a crucial role in ensemble learning as they form the fundamental components that work together to make collective predictions. The ensemble's final prediction is typically a combination of the predictions made by the base learners, either through voting, averaging, or weighted aggregation.

It is important to note that the success of an ensemble often relies on the diversity of the base learners. If all base learners are too similar or produce similar predictions, the ensemble may not be able to improve performance significantly. Therefore, the selection and diversity of base learners are crucial considerations when designing effective ensemble models.

Multinomial Logistic Regression

Multinomial logistic regression (MLR), also known as softmax regression, is a statistical regression model used for predicting categorical outcomes with three or more classes. It is an extension of binary logistic regression, which is used for binary classification problems. MLR is particularly useful when the dependent variable has more than two categories and the relationship between the independent variables and the outcome is assumed to be linear.

In MLR, the goal is to estimate the probabilities of each category of the dependent variable. The model assumes that the probabilities follow a multinomial distribution. The predicted probabilities are obtained using the softmax function, which transforms the linear combination of the independent variables into a probability distribution.

The model is typically trained using maximum likelihood estimation (MLE), where the parameters are estimated by maximizing the likelihood of observing the given data. The likelihood is calculated as the product of the predicted probabilities for the observed outcomes. The optimization process involves iteratively adjusting the parameters to find the values that maximize the likelihood.

Interpreting the results of MLR involves examining the estimated coefficients (also known as log odds or logits). These coefficients indicate the direction and magnitude of the relationship between the independent variables and the outcome categories. The exponential of the coefficients represents the odds ratios, which quantify the change in odds for a one-unit increase in the corresponding independent variable, holding other variables constant.

MLR can handle both categorical and continuous independent variables. Categorical variables are typically converted into dummy variables, with one category serving as the reference category. The interpretation of the coefficients in this case is in relation to the reference category.

It is important to assess the goodness of fit of the MLR model. This can be done using various measures, such as the likelihood ratio test, AIC (Akaike Information Criterion), or BIC (Bayesian Information Criterion). These measures help evaluate how well the model fits the data and compare different models.

MLR has applications in various fields, including social sciences, healthcare, marketing, and natural language processing. It is commonly used for tasks such as sentiment analysis, document classification, customer segmentation, and predicting multiple categories in survey data.

As the dataset has 5 different product IDs the MLR relies on a posterior class distribution, using Bayesian inference, to provide a *degree of plausibility* for each classification. The general equations of the MLR model are shown in equations 4.3 and 4.4, where: p_i is the probability of occurrence of each event; θ is the likelihood parameter; p_{k+1} represents the monotonicity of the lower bound iterate; $x = (x_1, \ldots, x_m)^T$ is the covariate-vector; and θ^i is the parameter vector corresponding to the $i-th$ response category (Böhning 1992; J. Li, Bioucas-Dias, and Plaza 2010).

$$p_i = \frac{\exp\left(\theta^{(i)^T} x\right)}{1 + \sum_{j=1}^{k} \exp\left(\theta^{(j)^T} x\right)} \text{ for i = 1, ..., k} \tag{4.3}$$

$$p_{k+1} = \frac{1}{1 + \sum_{j=1}^{k} \exp\left({\theta^{(j)}}^T x\right)} \tag{4.4}$$

Decision tree classifiers

Decision tree classifiers (DTC) are a rapid and useful top-down greedy approach to classify a dataset with a large number of variables (Farid et al. 2014). In general, each decision tree is a rule set. It is a supervised machine learning model that utilizes a tree-like structure to make predictions or classify instances based on a set of input features. They are popular due to their simplicity, interpretability, and ability to handle both categorical and numerical data.

The DTC starts with a root node that represents the entire dataset. It then recursively splits the data into subsets based on the values of different features, creating internal nodes and branches. Each internal node represents a decision based on a specific feature, and each branch represents a possible outcome or value of that feature. The process continues until a stopping criterion is met, such as reaching a maximum depth, a minimum number of instances in a leaf node, or no further improvement in splitting the data.

To make a prediction with a DTC, an instance is traversed down the tree from the root node to a leaf node. At each internal node, a decision is made based on the feature value, and the traversal continues until a leaf node is reached. The leaf node provides the predicted class or outcome for the instance.

The splitting of nodes in a decision tree is typically based on metrics that measure the homogeneity or impurity of the subsets created by the split. Common impurity measures include Gini impurity and entropy. The goal is to find splits that maximize the homogeneity within the subsets and minimize the impurity, resulting in pure leaf nodes.

DTC has several advantages. They are easy to understand and interpret, as the tree structure provides a clear visualization of the decision-making process. Decision trees can handle both numerical and categorical features, making them versatile for a wide range of datasets. They are also robust to outliers and can handle missing values by creating surrogate splits.

Researchers have used the ID3 (Iterative Dichotomiser) algorithm widely where information content is used to measure the attributes (Chandra and Varghese 2009). In the proposed approach, the handcrafted features were used to calculate the information content and then the classes were predicted. In addition to the decision tree and ML classifier, the CNN predictions described in section 4.1 were also included as part of the input layer to the neural network shown in figure 4.2.

4.3.2 Meta Learning using Stacking

Stacking, also known as stacked generalization, is a meta-learning technique that involves training a meta-learner or a higher-level model to make predictions based on the outputs of multiple base learners. It is a type of ensemble learning where the base learners' predictions are combined to create a new set of features that serve as input to the meta-learner.

Stacking offers several advantages over traditional ensemble methods. It allows for more complex relationships and interactions between the base learners, as the meta-learner can learn from the combined predictions and make a more informed decision. It can capture different perspectives and patterns present in the base learners' outputs, potentially improving the ensemble's performance.

Moreover, stacking can effectively handle situations where the base learners have complementary strengths or weaknesses. If certain base learners specialize in specific aspects of the data, the meta-learner can learn to leverage this diversity and make more accurate predictions.

In this experiment, the predictions from the base learners comprise layer 3 of the architecture. The predictions $y_{1i}, y_{2i}, ..., y_{4k}, y_{5k}$ are shown where i*(s)* are the predictions of MLR, j*(s)* are the predictions from DTC and k*(s)* are the predictions from CNN. These are then used in conjunction with the product weights with an additional learner neural network (layer 4) and the final predictions, *z1, z2, ..., z5* are delivered at layer 5 in the architecture.

4.4 Transfer Learning

Transfer learning is a machine learning technique that leverages knowledge learned from one task or domain to improve performance on a different but related task or domain. In the context of deep learning and neural networks, transfer learning involves using pre-trained models as a starting point for new tasks instead of training from scratch.

The process of transfer learning typically involves two main steps: pre-training and fine-tuning. In the pre-training phase, a deep model is trained on a large-scale dataset that is often unrelated to the target task. For instance, in computer vision, models may be trained on datasets like ImageNet, which contains millions of labeled images spanning a wide range of categories. During this phase, the model learns general features and representations that can be applied to various tasks.

Once the pre-training phase is complete, the knowledge acquired by the model can be transferred to the target task. In the fine-tuning phase, the pre-trained model is further trained on a smaller dataset that is specific to the target task. The idea is to adapt the learned features to the target task by adjusting the model's parameters while retaining the valuable knowledge obtained during pre-training. Fine-tuning often involves modifying the architecture of the pre-trained model, such as replacing or retraining the top layers to suit the target task.

Transfer learning offers several advantages. Firstly, it reduces the need for large amounts of labeled data, which can be expensive and time-consuming to acquire. By utilizing a pre-trained model as a starting point, transfer learning allows the target task to benefit from the general features learned from the source task. This is particularly beneficial when the target dataset is small or when there is a scarcity of labeled examples.

Additionally, transfer learning can significantly speed up the training process. Since the initial layers of the pre-trained model are frozen during fine-tuning, the model can converge faster on the target task. The lower layers, which capture low-level features like edges or textures, have already learned general representations that are likely to be useful across tasks. By reusing these representations, the model requires less time to learn the specific task-related features.

Transfer learning is applicable in various scenarios. One common scenario is domain adaptation, where the source and target domains are different but related. For example, a model trained on a dataset of indoor scenes can be transferred to a target task involving outdoor scenes. By leveraging the learned features from the source domain, the model can generalize well to the target domain, even with limited labeled data.

Another scenario is task adaptation, where the source and target tasks are related. For instance, a model trained for image classification can be used as a starting point for object detection or image segmentation tasks. By fine-tuning the model on the target task, it can learn to localize objects or segment regions of interest while still benefiting from the pre-trained knowledge of general image features.

The choice of a pre-trained model depends on the similarity between the source and target tasks or domains. In computer vision, popular pre-trained models include VGGNet, ResNet, InceptionNet, and EfficientNet, trained on large-scale datasets like ImageNet. These models have learned rich representations of images that can be effectively transferred to various computer vision tasks. In natural language processing, pre-trained models such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer) have revolutionized language understanding.

Approaches such as a Residual Network (ResNet) have been found to be successful in classifying images (K. He, X. Zhang, et al. 2016; Setyono, Chahyati, and Fanany 2018; Marsden et al. 2017). A ResNet is a CNN with a *skip connection*, which is also known as an identity shortcut connection. The concept behind the skip connection is to allow gradients to flow between layers as they help to reduce the impact of the vanishing gradient problem in deep learning architectures (K. He,

X. Zhang, et al. 2016). The general form is shown in equation 4.5, where θ is the learning parameter, $i = 1, 2, ..., m$ and $j = 0, 1, ..., m - 1$.

$$a^{(l+2i)} = g(\theta + a^{(l+2j)}) \tag{4.5}$$

For this research, a 34-layer ResNet architecture was used as it is well-balanced and gives accuracy equivalent to the CNN used in section 4.1 with relatively low computational power requirements (K. He, X. Zhang, et al. 2016).

4.5 Summary

This research involves the utilization of different deep learning and machine learning algorithms both individually and in combination. Specifically, a three-layer CNN architecture with a dropout layer has been employed. Additionally, in the parallel CNN approach, the weight is concatenated with the initial layer. For transfer learning purposes, ResNet 50 has been implemented. As for the base learners, Multinomial Logistic Regression (MLR) and Decision Tree Classifier (DTC) have been utilized in conjunction with CNN and a neural network serves as the meta-learner in the Ensemble approach.

Chapter 5

Experiment And Results

The primary purpose of the evaluation was to investigate the efficacy of the 4 machine learning techniques presented in Chapter 4 using the data acquired by the 4-step process outlined in Chapter 3.

5.1 Experimental Setup

Two broad sets of experiments were carried out in order to better understand the effect of a data transformation step on the predictive performance of the applied algorithms. In the first set of experiments, the colored input images are transformed to greyscale which has been shown to reduce the noise-to-signal ratio (Vidal and Amigo 2012), thus reducing the complexity and improving the performance of statistical learning techniques. In the second set of experiments, the color of the input images was primarily retained because it was hypothesized that the color contrasts which exist between the fat and meat components of each cut contained potentially useful information that would inform a better predictive performance. In each experiment, the datasets were split into a training set and a test set using an 80:20 stratified sampling ratio. The training set was further split using a 90:10 ratio for the purpose of implementing a validation strategy. The training data was used to train the model while the validation data was used to examine if the hyperparameters required further tuning. A hyperparameter is a parameter whose values cannot be estimated from the data and are external to the model. The test data was used as an unseen dataset to examine the results of the model.

The evaluation metrics used in image identification are typically accuracy, precision, recall, F1- score, and convergence time (Al-Sarayreh et al. 2018; Ropodi et al. 2015; Larsen et al. 2014; X. Yu et al. 2018; R. Wang, W. Li, and L. Zhang 2019; Setyono, Chahyati, and Fanany 2018). Although different methods have been used in a variety of studies, average accuracy is the most frequently used. In table 5.1, the results for the accuracy and the F1-score are shown for the training and test datasets, for each model, and for both the color and greyscale images. In addition, the convergence times for the color and greyscale images, for each method are also shown.

$$\text{Overall Accuracy} = \frac{\sum_{i=1}^{n=5} \text{TP}_i}{\text{N}} \tag{5.1}$$

In equation 5.1, TP_i or the True Positive is the number of instances predicted correctly for class i and N is the total number of predictions.

$$\text{F1}_i = 2.\frac{\text{Precision}_i * \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{5.2}$$

where

$$\text{precision}_i = \frac{TP_i}{\text{TP}_i + \text{FP}_i} \tag{5.3}$$

$$\text{recall}_i = \frac{TP_i}{\text{TP}_i + \text{FN}_i} \tag{5.4}$$

and FN_i and FP_i are the False Negative and False Positive respectively for product i.

The $F1_i$ Score, shown in equation 5.2, is a metric that combines both precision and recall for each class i. In comparison to accuracy, it is a useful metric as it penalizes the incorrectly classified minority sampled classes and is commonly reported as a weighted score of all the class F1 scores (Goutte and Gaussier 2005).

5.1.1 Experiment 1: CNN

In this set of experiments, the pre-processed greyscale and colored images were used as inputs in the CNN algorithm described in 4.2.1. The train accuracy, test accuracy, and the F1 score for B/W and color images are shown in the table 5.1.

5.1.2 Experiment 2: CNN with Product Weights.

In this experiment, the product weights were concatenated with the images used in the previous experiment as a second input within the flattened layer. The average accuracy and F1 score on the training and test dataset are shown in table 5.1.

5.1.3 Experiment 3: Ensemble Approach with Product Weights.

In this experiment the features outlined in 4.3 were used as input variables for both the Multinomial logistic regression and Decision Tree Classifiers while the pre-processed greyscale and the colored images were used as inputs for the CNN. These three algorithms were the level 1 models and acted as base learners Figure 4.2. At level 2, the predictions from these models have been used with the actual weights (in kgs) of the meat cuts as input for the neural network which acts as a meta learner. The result obtained with the greyscale and the colored images has been shown in the evaluation table 5.1.

5.1.4 Experiment 4: Transfer Learning - ResNet.

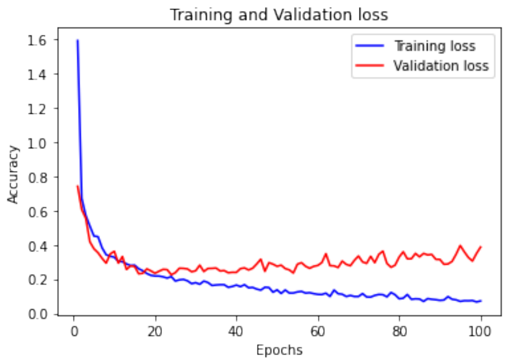
In this experiment, a 34-layer residual network with a weight initialization, was used (K. He, X. Zhang, et al. 2016). Again the network was fed the pre-processed greyscale and color images. The train and test accuracy and the F1 score have been shown in table 5.1.

5.2 Results

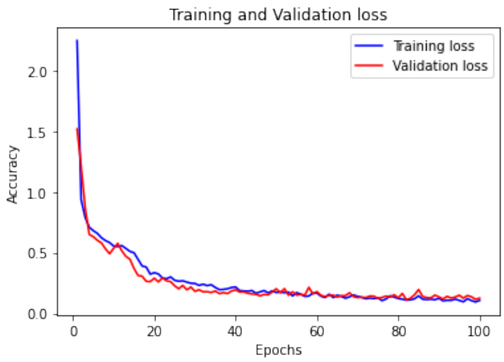
Accuracy statistics for each model and for both the color and grayscale images are in Table 2 for the training and test datasets. In addition, the convergence times for the color and grayscale images, for each method are also summarized in Table 5.1. While there was a wide disparity in convergence times, ranging from 4,025 seconds for the CNN on color images to 19,224 seconds for the Ensemble approach with color images, it was not unexpected given the difference in model complexities.

In order to determine the statistical significance of the results, a beta regression model with a "loglog" link function was implemented in the R programming language to model accuracy against the algorithm, dataset, and product variables. Only 2-way interaction terms on combinations of the product, algorithm, and image type were examined as the degrees of freedom in this particular analysis was limited to 40. The final beta regression model had a pseudo R2 of 0.98 and the comparison with an identity link was significant (ϕ=350.37, z=3.99, p<0.001). A Type III analysis was conducted and interaction effects between the algorithm and image type and between algorithm and product were found to be significant (Algorithm*Image Type F4,26 = 3.046 and P =0.016, Algorithm*Product F12,26 = 5.082 and P <0.001). From this analysis, a post-hoc analysis on the estimated marginal means with a Tukey correction for multiple comparisons was conducted and is outlined in Table 3.

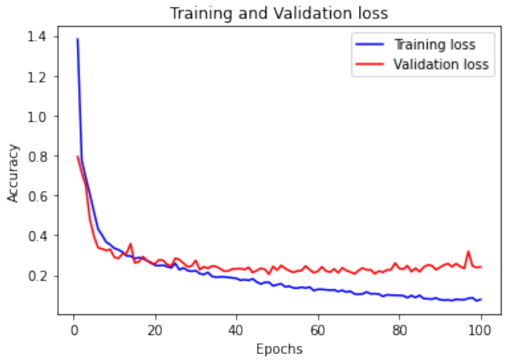
The Ensemble approach with color images was the best-performing algorithm with a test accuracy of 99.13% and a training accuracy of 99.50%. The estimated marginal mean (EMM) for the test accuracy difference on color images was higher for the Ensemble approach compared with either the CNN ((EMMCNN-EMMEnsemble) Zscore= -4.72 or P < 0.001) the ResNET ((EMMEnsemble-EMMResNET) Z score= 7.82 or P < 0.001) algorithms without incorporating the cut weight information. The same algorithm also performed best for images in grayscale, with a test accuracy score of 95.00% and a train accuracy of the same value. However, the only statistical difference found was between the Ensemble and the ResNET without



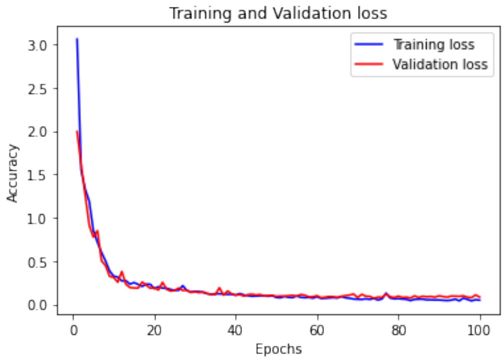
(a) CNN without weights B/W overfit



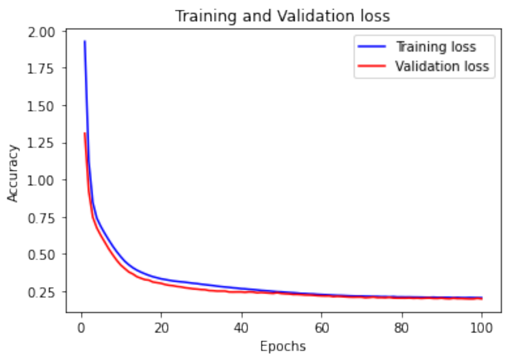
(b) CNN without weights color no overfit



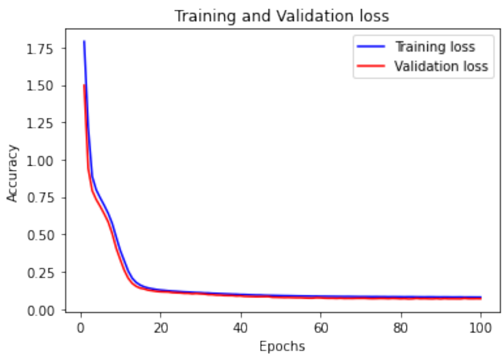
(c) CNN with weights B/W overfit



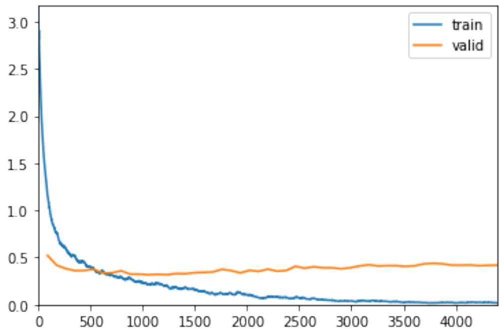
(d) CNN with weights color no overfit



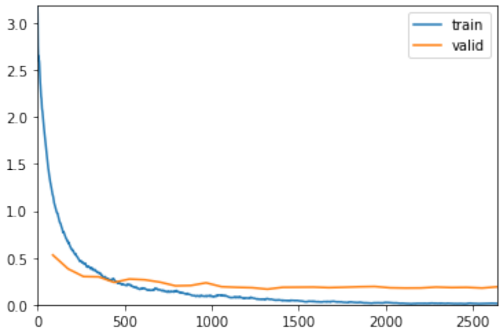
(e) Ensemble B/W no overfit



(f) Ensemble color no overfit



(g) ResNet B/W overfit



(h) ResNet color overfit

Figure 5.1: Training and Validation Loss Graphs.

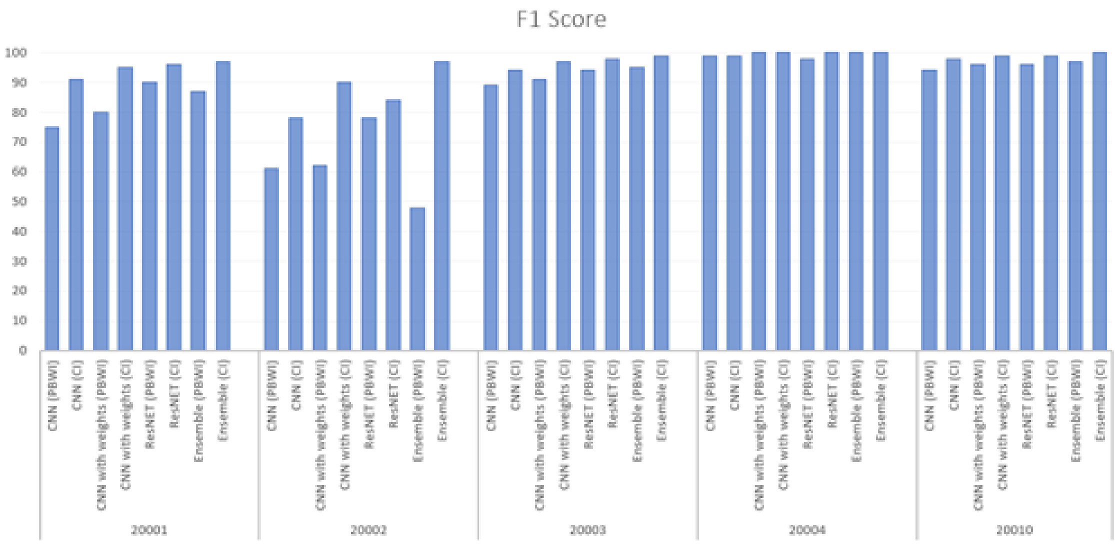


Figure 5.2: F1 score for all five meat cuts with different models on both the pre-processed black and white (PBWI) and the colored images (CI)

using cut weight information algorithms ((EMMEnsemble-EMMResNET) Z score= 4.42 or P $<$ 0.001). With a score of 98.00%, the Ensemble approach also had the highest weighted-average F1 score. The weighted-average F1 score was derived from the average F1 score from each classification category weighted by the number of meat cuts in each product group.

Figure 5.1 illustrates both the training and validation accuracy as the number of epochs changed for each method, for both the color and grayscale images. All approaches, with the exception of the Ensemble approach, demonstrated varying degrees of percentage difference in accuracy between the training and test accuracy on the grayscale images (CNN 4.80%, CNN with weights 5.80% and ResNET 0.90% and Ensemble 0.00%), implying the algorithms over-fitted the training data. The level of overfitting was reduced for both the CNN and the CNN that also used the cut weight information, albeit, there was a marginal increase in overfitting with the ResNET and Ensemble approaches for the color images (CNN 2.90%, CNN with weights 1.60% and ResNET 1.30% and Ensemble 0.43%).

All five algorithms, CNN, CNN concatenated with weights, ResNET, ResNET concatenated with weights and the Ensemble method performed better with color images, as the EMM difference between algorithms run on color images with those run on grayscale images was statistically significant ((EMMcolour-EMMgrayscale) Z ratio = 13.649, $P < 0.001$). This was an interesting finding as the color images did not receive any pre-processing.

The inclusion of product weights in the model demonstrated a beneficial effect when detecting meat cuts from images, as the CNN and the Ensemble approaches when including weights out-performed the same algorithms when excluding the weights ((EMMCNN with Weights-EMMCNN) Z ratio = 3.527, $P < 0.015$, ((EMM-CNN with Weights-EMMResNET) Z ratio = 5.37, $P < 0.001$, ((EMMEnsemble-EMMCNN) Z ratio = 3.211, $P < 0.043$, ((EMMEnsemble-EMMResNET) Z ratio = 5.095, $P<0.001$).

Figure 5.2 shows the F1 score for each model for each individual meat cut. In all cases, the highest F1 score was achieved for the Ensemble method with colored images (CI); while meat cut 20004 had the highest F1 score (100.00%) using the Ensemble method. Meat cut 20002, had the fewest number of images and correspondingly had the smallest F1 scores. However, using the Ensemble method with CI, meat cut 20002 did have the highest F1 score (97.00%).

In table 5.1 the accuracy for the training and test datasets, and the weighted-average F1 score for the test dataset are shown in the columns $Train\ Accuracy$, $Test\ Accuracy$, and $Test\ Weighted\ F1\ Score$ respectively. For each of the four models, there are two rows representing the pre-processed black and white images (PBWI) and the colored images (CI). The time(S) column displays the time in seconds, taken for the experiment to train the model.

Table 5.1: Comparative Performances for all 4 Models

Model	**Image type**	**Train Accuracy**	**Test Accuracy**	**Average Precision (Test)**	**Average Recall (Test)**	**Weighted F1 Score (Test)**	**Time (S)**
CNN	PBWI	96.80%	92.00%	86.00%	82.00%	84.00%	6675
CNN	CI	98.90%	96.00%	96.00%	92.00%	92.00%	3093
CNN with Weights	PBWI	98.80%	93.00%	91.00%	83.00%	86.00%	6059
CNN with Weights	CI	99.60%	98.00%	98.00%	95.00%	96.00%	11251
ResNet	PBWI	91.80%	90.90%	90.90%	90.80%	90.80%	1745
ResNet	CI	96.80%	96.50%	96.50%	96.00%	96.00%	12500
Ensemble	PBWI	95.00%	95.00%	92.00%	82.00%	85.00%	18518
Ensemble	CI	99.50%	99.13%	99.00%	98.00%	98.00%	19224

Table 5.2: Post hoc contrast analysis of predicted Marginal mean difference between algorithms by image type

Image Type	Contrast	Marginal Mean Difference	SE	Z ratio	P value
Color	CNN-CNN with weights	-0.7573	0.215	0.0153	0.0153
Color	CNN-Ensemble	-0.6719	0.209	-3.211	0.0433
Color	CNN-ResNET	0.3505	0.174	2.017	0.5871
Color	CNN with weights-Ensemble	0.0584	0.237	0.361	1
Color	CNN with weights-ResNET	1.1078	0.206	5.37	<0.001
Color	Ensemble-ResNET	1.0224	0.201	5.095	<0.001
Grayscale	CNN-Ensemble	-0.4688	0.175	-2.687	0.1789
Grayscale	CNN-ResNET	0.2886	0.145	1.996	0.602
Grayscale	CNN with Weights-Ensemble	-0.3269	0.183	-1.782	0.7468
Grayscale	CNN with Weights-ResNET	0.4306	0.155	2.769	0.147
Grayscale	Ensemble-ResNET	0.755	0.171	4.422	0.0004

5.3 Summary

The primary aim of this study was to create an automated meat cut identification strategy for beef boning lines that simultaneously process multiple beef cuts; the present study focused solely on the cuts from the Semimembranosus muscle. In order to do this, a number of state-of-the-art image detection methodologies and a novel Ensemble strategy were applied to a dataset consisting of 7987 product cut images and their corresponding weights. A series of eight experiments were conducted on both color and preprocessed grayscale images and the novel Ensemble approach developed in this study performed best for each individual cut using color images and outperformed those using grayscale while availing of product weights and also improved the accuracy of categorization. These results demonstrated some interesting findings relating to AI and implementation strategies for future commercial deployment strategies.

Chapter 6

Discussion

The primary aim of this study was to create an automated meat cut identification strategy for beef boning lines that simultaneously process multiple beef cuts; the present study focused solely on the cuts from the semimembranosus muscle. In order to do this, a number of classical neural networks that perform image detection and a novel Ensemble strategy were applied to a dataset (McCarren, Scriney, et al. 2021) consisting of 7,987 product cut images and their corresponding weights. A series of eight experiments were conducted on both color and preprocessed grayscale images, and the novel Ensemble approach developed in this study performed best for each individual cut and that using color images outperformed those that used grayscale while availing of product weights also improved the accuracy of categorization. These results demonstrated findings relating to artificial intelligence (AI) and implementation strategies that would be applicable for future commercial deployment strategies.

6.1 AI Strategy

Typically, in image detection problems, one highlights image features using a variety of pre-processing techniques to improve the algorithm's performance. However, in the live production environment, where these experiments were conducted, the opposite result was found; accuracy and weighted-average F1 score was 4.00% higher for all models using color images. While this is not typical in object detection problems (Y. Xu et al. 2016), the occurrence in these experiments can be explained by the fact that the background remained relatively constant throughout the experimental period, thus removing it from the images had little or no effect. In addition, grayscaling the images potentially limited the ability of all algorithms to differentiate between fat and red meat.

6.1.1 Meat Cuts and Product Weight

In the meat industry, meat cuts are generally extracted from primal cuts, and knowing the weights of these cuts can potentially help in the identification of candidate labels. Results from the present study clearly demonstrate the benefit of knowing the weight of the on-coming cut, as the inclusion of the product weight into the flat layer of both the CNN and ResNET improved the resulting meat cut identification. This is not surprising as it has been shown to be successful in previous research on product identification (Shi et al. 2020). However, in this study, a simplified model where product weights alone were used as the only independent variable resulted in an accuracy of 60.12% on the test dataset. This result justifies the importance of the product weights but also demonstrates that the product weights alone are not sufficient for categorizing product cuts.

6.1.2 The Role of Transfer Learning and Algorithm Selection

Transfer learning is one of the more recent evolutions of machine learning and, in particular, the ResNET transfer learning algorithm is considered to be one of the most advanced deep learning architectures in image detection (Marsden et al. 2017). However, in the experiments conducted in the present study, the incorporation of the weight of each meat cut in the final layer and the outputs of the simpler approaches outperformed the ResNET architecture. While this was somewhat surprising, the combined use of multinomial logistic regression, the CNN and the decision tree algorithm in the ensemble approach on the set of artificially created features, was the most consistent with respect to overfitting and suggests that the use of simpler algorithms in the Ensemble approach may have assisted the CNN algorithm in finding a stable solution. While the Ensemble approach with color images required took longer to converge, the ability to avoid overfitting is extremely important in a live environment.

6.1.3 Overcoming Overfitting and Ensuring Stability

In a live environment, the convergence time would not be a considerable issue as model fitting would only be implemented in order to calibrate the model in an offline mode. Finding a stable solution can be an issue when using Neural Network algorithms as the level of non-linearity in the cost function can cause overfitting (Nguyen et al. 2011). Using a mixture of simpler algorithms in the early stage of the Ensemble has been shown to outperform more complex methods with regard to accuracy and F1-score (Abdelaal et al. 2018) and to reduce overfitting (Perrone and Cooper 1995). (Gc, Y. Zhang, et al. 2021) achieved a maximum test accuracy of 98.57% and a weighted average F1-score of 94.00% on the test dataset of beef cuts using the alternative VGG16 transfer learning model, a state-of-the-art method. The proposed Ensemble method was able to achieve an accuracy up to 99.13% and a weighted-average F1-score of 98.00%. While this improvement in accuracy appears modest, it effectively reduces the error by over 39.00

6.2 Deployment Strategy

The data capture unit developed in the present study was implemented using the Node.js programming language, and consisted of a DEM weighing scale (System 1985), a DEM terminal, and a Vivotek harsh environment camera. In order to truly automate the collection of the cut weight and subsequently identify the products in a live environment, an external harsh environment color camera will need to be integrated into an inline weighing scale. The terminal for this scale will then need a script that runs the Ensemble machine learning models; however, the code used to create the Ensemble approach in the present study can be easily integrated into many diverse operating systems. For each new group of products, the algorithm will need to be trained on images collected from the live production of the corresponding plant.

The number of samples required to train the algorithm will be problem specific. However, in previous research studies, researchers have recommended that at least 1000 images of each object should be used during the AI training phase (Cho et al. 2015). This is not a hard rule and in this study, the results demonstrated that there was ample data with the exception of product 20002, where the overall accuracy was lower. As mentioned previously, the data collection for this study was implemented on bespoke software. This code can be readily implemented to help create training data for the Ensemble machine learning algorithm during new deployments and makes the implementation in a commercial environment an attractive proposition.

6.2.1 Cost-Effectiveness

The deployment of the system in a live environment is not expected to be expensive, as all the software used is open source. The Node.js programming language and Python (Tilkov and Vinoski 2010; Python 1991), which are both open source, were employed in the development of the data capture unit. The camera technology used in the study was also relatively inexpensive, as the image processing did not require the use of spectral images. This finding is noteworthy, as previous studies (Larsen et al. 2014; Ropodi et al. 2015; Al-Sarayreh et al. 2018; X. Yu et al. 2018) have often relied on multispectral or infrared spectroscopy for image processing in similar applications. The advancements in object detection algorithms and the incorporation of weights in this study have potentially eliminated the need for expensive infrared spectroscopy, opening up possibilities for cost-effective implementation in various food industry applications.

6.2.2 Accuracy of the Ensemble Algorithm

The results of the study highlight the ability of artificial intelligence (AI) to replicate the behavior of a human operator. The test accuracy achieved with the Ensemble algorithm demonstrates the effectiveness of AI in accurately identifying and categorizing products. By incorporating various algorithms, including multinomial logistic regression, CNN, and decision tree, the Ensemble approach outperformed more complex deep learning architectures like Residual Neural Network (ResNET). This finding emphasizes the importance of algorithm selection and the potential benefits of combining simpler algorithms in ensemble models, especially in live environments where overfitting and stability are crucial.

6.3 Applications

6.3.1 The Need for Automation in Beef Boning Process

The meat processing industry often considers the implementation of automated or robotic processes based on the potential return on investment. Improved product quality, reduced labor costs, and a decrease in safety incidents are some of the factors influencing this decision (Purnell and Further 2013; Caldwell 2012). Automation has already found its way into the sector, with applications such as fat and red meat yield prediction and a limited number of cutting procedures being automated (Pabiou et al. 2011). However, the beef boning process still heavily relies on manual labor, particularly on modern pace boning lines. In these operations, operators are responsible for identifying products, checking their quality characteristics, and redirecting them to the appropriate packing stations. Currently, there is no provision for monitoring yields during the boning process when multiple cuts are being processed simultaneously. This poses a significant challenge, as plant management relies on continuous supervision to monitor the boning operators' cut decisions. By automating the identification of meat cuts and incorporating automated weighing technology, it becomes possible to accurately monitor the yield of each cut relative to the original primal weight during production, resulting in improved meat yield for the plant.

Apart from the potential yield improvement, the elimination of an operator from the production line can also reduce the risk of cross-contamination from bacteria such as Staphylococcus or Escherichia coli, which are commonly transmitted by line operators during food operations (Véronique 2008). However, the absence of a trained human operator may increase the likelihood of misspecification of meat cuts. To mitigate this issue, the system applied in this study can be adapted to remove products onto a separate quality control (QC) line if the system fails to recognize the meat cut or if it falls outside the weight specifications. This approach effectively mimics the actions of a human operator, ensuring that quality control measures are maintained throughout the automated process. By incorporating such safeguards, the system can maintain high standards of accuracy and minimize the risk of misspecification, thereby enhancing food safety protocols.

6.3.2 Real-time Monitoring for Improved Performance

Implementing automated weighing technology and integrating it with the identification of meat cuts offers several advantages beyond improved yield and reduced contamination risks. One significant advantage is the ability to monitor and assess the performance of the boning process in real-time. Traditional systems rely on end-of-batch assessments, which may result in delays in identifying and rectifying any issues. By monitoring yields and other quality characteristics during production, plant management can promptly address any deviations, ensuring that production remains on track and consistently meets the required specifications. Real-time monitoring also enables more effective planning and scheduling, as adjustments can be made as soon as deviations are detected, minimizing disruptions and optimizing production efficiency.

The integration of automation and robotic processes into the meat processing industry has the potential to revolutionize the sector by streamlining operations and enhancing overall productivity. However, it is crucial to strike a balance between automation and human expertise to ensure optimal outcomes. While automation offers numerous benefits, the human factor cannot be entirely replaced, particularly when it comes to complex tasks such as meat cut identification. In this study, the system applied successfully combined automation with the ability to replicate the decision-making capabilities of human operators, ensuring the accuracy and reliability of the process. By utilizing advanced technologies and integrating human-like decision-making algorithms, the meat processing industry can leverage the advantages of automation while preserving the critical aspects of human expertise.

In the meat processing industry, the decision to implement automated or robotic processes is usually dictated by the return on investment which, in turn, is usually a function of improved product quality, reduced labor costs, or a reduction in safety incidents (Purnell and Further 2013; Caldwell 2012). Automation has been introduced in the sector and has been used in applications such as fat and red meat yield prediction (Pabiou et al. 2011) and a limited number of cutting procedures. However, beef boning is still predominantly a highly manual process on modern pace boning lines. These operations rely on operators at the end of the line to identify products, check their quality characteristics, and then manually redirect them to the appropriate packing stations. At present, in operations where there are multiple cuts being processed simultaneously, there is generally no facility to monitor yields during the boning process. This is a major weakness in current systems as plant management relies on in-line supervision to continually monitor the operator cut decisions of boning operators. By automating the identification of the relevant meat cuts and, in conjunction with automated weighing technology, the yield of the cut relative to the original primal weight can be accurately monitored during production rather than at the end of the batch, thus improving the meat yield of the plant.

6.4 Summary

The implementation of automated weighing technology and automated meat cut identification systems presents significant opportunities for improving the meat processing industry. By monitoring yields and quality characteristics in real-time, plant management can enhance production efficiency, reduce labor costs, and ensure consistent product quality. The removal of operators from the production line also minimizes the risk of cross-contamination, enhancing food safety.

Chapter 7

Conclusion And Future Work

In this final chapter, the concluding remarks of the thesis have been presented, outlining the contributions, acknowledging any constraints in this work, and proposing avenues for future research. The chapter is structured as follows: Section 7.1 provides a summary of the main discoveries and insights obtained from each technical chapter. In Section 7.2, the limitations encountered during this study have been discussed by identifying potential directions for future investigations.

7.1 Thesis Summary

Access to a skilled workforce is crucial for industries that rely on human involvement in their production processes. The meat industry is one such sector, and the importance of skilled labor was evident during the COVID-19 pandemic when absenteeism levels were high. While some processes in the meat industry have been partially automated, such as meat cutting and fat determination, the labeling and identification of meat cuts still heavily rely on human intervention and manual handling. This reliance on manual labor can lead to increased costs and the potential for errors and microbiological contamination. Automating the labeling and identification of meat cuts could help address these challenges. By implementing automated systems, businesses in the meat industry can reduce labor costs, minimize errors, and mitigate the risk of microbiological contamination. Furthermore, automation can enhance productivity and efficiency in meat processing, ensuring consistent and accurate identification of meat cuts.

In this thesis, an approach to automate the identification of meat cuts using a live beef production line over a three-week period has been proposed. It was unclear at the outset as to which machine learning model would perform best on these types of images in the live environment and thus a number of computer vision algorithms were evaluated. As is normal with the construction of a new dataset, imbalances in terms of image distribution frequencies can occur but this was offset using different pre-processing methods and data augmentation.

The experimental studies conducted in the meat industry shed light on the importance of considering color images, product weights, and algorithm selection in image detection tasks. The incorporation of weight information significantly enhances the identification of meat cuts, while the ensemble approach, utilizing a combination of simpler algorithms and artificial features, outperforms complex deep learning architectures. By understanding these insights and employing suitable techniques, the meat industry can further improve its image detection capabilities, enabling more accurate categorization and quality control processes.

This also showcases the development and implementation of a data capture unit using the Node.js programming language, along with a DEM weighing scale and a Vivotek harsh environment camera. The integration of an external color camera with an inline weighing scale, along with the implementation of the Ensemble machine learning models, holds promise for fully automating the collection of cut weights and product identification in live environments. The availability of open-source software and the elimination of the need for costly spectral images through advancements in object detection algorithms make the system cost-effective and applicable to a range of food industry applications. The accuracy achieved by the Ensemble algorithm further validates the potential of artificial intelligence in replicating human operator behavior and improving overall efficiency in industrial processes.

7.2 Limitation and Future Work

This study has the following limitations:

- Limited Dataset: One limitation of this study is the relatively small dataset used for training and evaluation. The dataset primarily focused on a specific range of primal cuts, which may limit the generalizability of the results to a broader set of meat products. To address this limitation, future work should involve constructing a larger dataset with a more diverse range of primal cuts to ensure the robustness of the model.

- Scope of Evaluation: The evaluation of the models was conducted on a specific test dataset, which may not fully represent the variability and complexity encountered in a real-world commercial application. Further testing and validation on a more challenging dataset, possibly encompassing different production environments and variations in product characteristics, would provide a more comprehensive assessment of the model's performance.

The potential future work can be:

- Expansion of Dataset: To enhance the applicability of the developed model, future work should involve collecting a larger and more diverse dataset. Including a broader range of primal cuts, variations in quality attributes, and potential challenges encountered in real-world scenarios would help improve the robustness and generalizability of the model.

- Real-world Implementation: The next step in this research is to apply the best-performing model in a full-scale commercial application. This involves integrating the model into an automated system that can accurately identify and label meat cuts in a live production environment. The performance and feasibility of the model should be evaluated under real-world conditions, considering factors such as lighting variations, noise, and production line dynamics.

- Model Optimization: Further optimization of the model can be explored to improve its accuracy, efficiency, and convergence rate. This may involve fine-tuning the hyperparameters, exploring different architectures or ensembles, and investigating advanced techniques such as transfer learning or data augmentation. By continually refining the model, its performance can be further enhanced, leading to more reliable and accurate meat cut identification.

- System Integration and User Interface: Future work should focus on integrating the developed model into a user-friendly system with an intuitive interface. This would facilitate its adoption and usage by industry professionals, allowing them to easily access and utilize the model for automated meat cut identification. User feedback and usability testing should be conducted to ensure the system meets the practical needs of the meat processing industry.

Bibliography

Turing, Alan Mathison (1950). “Mind”. In: *Mind* 59.236, pp. 433–460.

Tomek, I. (1976). “Two modifications of CNN. IEEE Trans”. In: *Systems, Man and Cybernetics* 6, pp. 769–772.

Bostian, Marianne L et al. (1985). “Automated methods for determination of fat and moisture in meat and poultry products: collaborative study”. In: *Journal of the Association of Official Analytical Chemists* 68.5, pp. 876–880.

System, DEM Integrated (1985). *Industrial weighing scales: Weighing scales ireland.* URL: `https://demmachines.com/product-category/weighing-scales/` (visited on 05/16/2023).

Python (1991). *Python release python 3.6.0.* URL: `https://www.python.org/downloads/release/python-360/` (visited on 04/16/2023).

Böhning, Dankmar (1992). “Multinomial logistic regression algorithm”. In: *Annals of the institute of Statistical Mathematics* 44.1, pp. 197–200.

Perrone, Michael P and Leon N Cooper (1995). “When networks disagree: Ensemble methods for hybrid neural networks”. In: *How We Learn; How We Remember: Toward An Understanding Of Brain And Neural Systems: Selected Papers of Leon N Cooper.* World Scientific, pp. 342–358.

Freund, Yoav and Robert E Schapire (1997). “A decision-theoretic generalization of on-line learning and an application to boosting”. In: *Journal of computer and system sciences* 55.1, pp. 119–139.

Breiman, Leo (2001). “Random forests”. In: *Machine learning* 45, pp. 5–32.

Chawla, N. V. et al. (2002). “SMOTE: synthetic minority over-sampling technique”. In: *Journal of artificial intelligence research* 16, pp. 321–357.

Vilalta, Ricardo and Youssef Drissi (2002). “A perspective view and survey of meta-learning”. In: *Artificial intelligence review* 18, pp. 77–95.

Shah, Satish and Vaibhav Gandhi (2004). “Image classification based on textural features using artificial neural network (ANN)”. In: *Journal of The Institution of Engineers (India): Series A* 84, pp. 72–77.

Weiss, Gary M (2004). “Mining with rarity: a unifying framework”. In: *ACM Sigkdd Explorations Newsletter* 6.1, pp. 7–19.

Goutte, Cyril and Eric Gaussier (2005). “A probabilistic interpretation of precision, recall and F-score, with implication for evaluation”. In: *European conference on information retrieval*. Springer, pp. 345–359.

Han, Hui, Wen-Yuan Wang, and Bing-Huan Mao (2005). “Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning”. In: *Advances in Intelligent Computing: International Conference on Intelligent Computing, ICIC 2005, Hefei, China, August 23-26, 2005, Proceedings, Part I 1*. Springer, pp. 878–887.

Sabol, John Michael et al. (Feb. 2006). *Method and apparatus for quantifying tissue fat content*. US Patent 6,999,549.

McCarthy, John (Nov. 2007). *WHAT IS ARTIFICIAL INTELLIGENCE?* URL: `https://www-formal.stanford.edu/jmc/whatisai/whatisai.html` (visited on 05/16/2023).

He, H. et al. (2008). “June”. In: *ADASYN: Adaptive synthetic sampling approach for imbalanced learning*. In 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence) . IEEE, pp. 1322–1328.

Véronique, COMA (2008). “Bioactive packaging technologies for extended shelf life of meat-based products”. In: *Meat science* 78.1-2, pp. 90–103.

Chandra, B and P Paul Varghese (2009). “Moving towards efficient decision tree construction”. In: *Information Sciences* 179.8, pp. 1059–1069.

Chen, K et al. (2010). “Color grading of beef fat by using computer vision and support vector machine”. In: *Computers and Electronics in Agriculture* 70.1, pp. 27–32.

Li, Jun, José M Bioucas-Dias, and Antonio Plaza (2010). “Semisupervised hyperspectral image segmentation using multinomial logistic regression with active learning”. In: *IEEE Transactions on Geoscience and Remote Sensing* 48.11, pp. 4085–4098.

Russell, Stuart J (2010). *Artificial intelligence a modern approach*. Pearson Education, Inc.

Tilkov, Stefan and Steve Vinoski (2010). “Node. js: Using JavaScript to build high-performance network programs”. In: *IEEE Internet Computing* 14.6, pp. 80–83.

Tola, E., V. Lepetit, and P. Fua (2010). “DAISY: An Efficient Dense Descriptor Applied to Wide-Baseline Stereo”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.5, pp. 815–830. DOI: 10.1109/TPAMI.2009.77.

Warriss, Paul D (2010). *Meat science: an introductory text*. Cabi.

Hviid, Marchen, Mikkel Engbo Jørgensen, and Anders Lindbjerg Dahl (2011). “Vision based meat tracking”. In: *58th International Congress of Meat Science and Technology*.

Inc., Apple (Oct. 2011). *Siri*. URL: https://www.apple.com/siri/ (visited on 05/16/2023).

Nguyen, Huu Minh et al. (2011). “An alternative approach to avoid overfitting for surrogate models”. In: *Proceedings of the 2011 winter simulation conference (WSC)*. IEEE, pp. 2760–2771.

Pabiou, T et al. (2011). “Use of digital images to predict carcass cut yields in cattle”. In: *Livestock Science* 137.1-3, pp. 130–140.

Umino, Tatsuya et al. (Dec. 2011). *Automatic deboning method and apparatus for deboning bone laden meat*. US Patent 8,070,567.

Underly, Kari (2011). *The Art of Beef Cutting: A Meat Professional's Guide to Butchering and Merchandising*. John Wiley & Sons.

Caldwell, Darwin G (2012). *Robotics and automation in the food industry: current and future technologies*. Elsevier.

Napierala, K. and J. Stefanowski (2012). "March. Identification of different types of minority class examples in imbalanced data". In: *International Conference on Hybrid Artificial Intelligence Systems . , , Heidelberg*. Berlin: Springer, pp. 139–150.

Vidal, Maider and José Manuel Amigo (2012). "Pre-processing of hyperspectral images. Essential steps before image analysis". In: *Chemometrics and Intelligent Laboratory Systems* 117, pp. 138–148.

Cantelon, Mike et al. (2013). "Node. js in Action". In.

Choi, S. et al. (2013). "Applications and requirements of industrial robots in meat processing". In: *2013 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 1107–1112. DOI: 10.1109/CoASE.2013.6653967.

Purnell, G and Grimsby Institute of Further (2013). "Robotics and automation in meat processing". In: *Robotics and Automation in the Food Industry*. Elsevier, pp. 304–328.

Corporation, Microsoft (Apr. 2014). *Microsoft*. URL: https://www.microsoft.com/en-us/cortana (visited on 05/16/2023).

Farid, Dewan Md et al. (2014). "Hybrid decision tree and naıve Bayes classifiers for multi-class classification tasks". In: *Expert systems with applications* 41.4, pp. 1937–1946.

Fricke, PM et al. (2014). "Expression and detection of estrus in dairy cows: the role of new technologies". In: *Animal* 8.s1, pp. 134–143.

Larsen, Anders Boesen Lindbo et al. (2014). "Vision-based method for tracking meat cuts in slaughterhouses". In: *Meat science* 96.1, pp. 366–372.

Menardi, G. and N. Torelli (2014). "Training and assessing classification rules with imbalanced data". In: *Data mining and knowledge discovery* 28.1, pp. 92–122.

Simonyan, Karen and Andrew Zisserman (2014). “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556*.

Srivastava, Nitish et al. (2014). “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1, pp. 1929–1958.

Cho, Junghwan et al. (2015). “How much data is needed to train a medical image deep learning system to achieve necessary high accuracy?” In: *arXiv preprint arXiv:1511.06348*.

Kingma, Diederik P. and Jimmy Ba (2015). “Adam: A Method for Stochastic Optimization”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun.

Ren, Shaoqing et al. (2015). “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems* 28.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, pp. 234–241.

Ropodi, AI et al. (2015). “Multispectral image analysis approach to detect adulteration of beef and pork in raw meats”. In: *Food Research International* 67, pp. 12–18.

Szegedy, Christian et al. (2015). “Going deeper with convolutions”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9.

Beltran, A et al. (2016). “Adapting the eButton to the abilities of children for diet assessment”. In: *Proceedings of Measuring Behavior 2016: 10th International Conference on Methods and Techniques in Behavioral Research. International Con-*

ference on Methods and Techniques in Behavioral Research (10th: 2016: Dublin, Ireland). Vol. 2016. NIH Public Access, p. 72.

He, Kaiming, Xiangyu Zhang, et al. (2016). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

LLC, Google (May 2016). *Hey Google.* URL: `https://assistant.google.com/` (visited on 05/16/2023).

Park, Sungheon and Nojun Kwak (2016). "Analysis on the dropout effect in convolutional neural networks". In: *Asian conference on computer vision.* Springer, pp. 189–204.

Redmon, Joseph, Santosh Divvala, et al. (2016). "You only look once: Unified, real-time object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788.

Xu, Yong et al. (2016). "Approximately symmetrical face images for image preprocessing in face recognition and sparse representation based classification". In: *Pattern Recognition* 54, pp. 68–82.

Zhao, Hang et al. (2016). "Loss functions for image restoration with neural networks". In: *IEEE Transactions on computational imaging* 3.1, pp. 47–57.

DeVries, Terrance and Graham W Taylor (2017). "Improved regularization of convolutional neural networks with cutout". In: *arXiv preprint arXiv:1708.04552.*

Granik, Mykhailo and Volodymyr Mesyura (2017). "Fake news detection using naive Bayes classifier". In: *2017 IEEE first Ukraine conference on electrical and computer engineering (UKRCON).* IEEE, pp. 900–903.

He, Kaiming, Georgia Gkioxari, et al. (2017). "Mask r-cnn". In: *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969.

Huang, Gao et al. (2017). "Snapshot ensembles: Train 1, get m for free". In: *arXiv preprint arXiv:1704.00109.*

Jiang, Fei et al. (2017). "Artificial intelligence in healthcare: past, present and future". In: *Stroke and vascular neurology* 2.4.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton (2017). "Imagenet classification with deep convolutional neural networks". In: *Communications of the ACM* 60.6, pp. 84–90.

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell (2017). "Simple and scalable predictive uncertainty estimation using deep ensembles". In: *Advances in neural information processing systems* 30.

Lin, W. C. et al. (2017). "Clustering-based undersampling in class-imbalanced data". In: *Information Sciences* 409, pp. 17–26.

Marsden, Mark et al. (2017). "ResnetCrowd: A residual deep learning architecture for crowd counting, violent behaviour detection and crowd density level classification". In: *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, pp. 1–7.

McCarren, Andrew, Suzanne McCarthy, et al. (2017). "Anomaly detection in agri warehouse construction". In: *Proceedings of the Australasian Computer Science Week Multiconference*, pp. 1–10.

Redmon, Joseph and Ali Farhadi (2017). "YOLO9000: better, faster, stronger". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271.

Shelke, Mayuri S, Prashant R Deshmukh, and Vijaya K Shandilya (2017). "A review on imbalanced data handling using undersampling and oversampling technique". In: *Int. J. Recent Trends Eng. Res* 3.4, pp. 444–449.

Vaswani, Ashish et al. (2017). "Attention is all you need". In: *Advances in neural information processing systems* 30.

Abdelaal, Hammam M et al. (2018). "Improve the automatic classification accuracy for Arabic tweets using ensemble methods". In: *Journal of Electrical Systems and Information Technology* 5.3, pp. 363–370.

Cai, Zhaowei and Nuno Vasconcelos (2018). "Cascade r-cnn: Delving into high quality object detection". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6154–6162.

Cubuk, Ekin D et al. (2018). “Autoaugment: Learning augmentation policies from data”. In: *arXiv preprint arXiv:1805.09501*.

Lancaster, Jenessa et al. (2018). “Bayesian optimization for neuroimaging pre-processing in brain age classification and prediction”. In: *Frontiers in aging neuroscience* 10, p. 28.

Law, Hei and Jia Deng (2018). “Cornernet: Detecting objects as paired keypoints”. In: *Proceedings of the European conference on computer vision (ECCV)*, pp. 734–750.

McAllister, Patrick et al. (2018). “Combining deep residual neural network features with supervised machine learning algorithms to classify diverse food image datasets”. In: *Computers in biology and medicine* 95, pp. 217–233.

Patrıcio, Diego Inácio and Rafael Rieder (2018). “Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review”. In: *Computers and electronics in agriculture* 153, pp. 69–81.

Prabhu, Raghav (2018). “Understanding of convolutional neural network (CNN)—deep learning”. In: *A Medium Corporation, US*.

Redmon, Joseph and Ali Farhadi (2018). “Yolov3: An incremental improvement”. In: *arXiv preprint arXiv:1804.02767*.

Al-Sarayreh, Mahmoud et al. (2018). “Detection of red-meat adulteration by deep spectral–spatial features in hyperspectral images”. In: *Journal of Imaging* 4.5, p. 63.

Setyono, Noer Fitria Putra, Dina Chahyati, and Mohamad Ivan Fanany (2018). “Betawi Traditional Food Image Detection using ResNet and DenseNet”. In: *2018 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. IEEE, pp. 441–445.

Vanschoren, Joaquin (2018). *Meta-Learning: A Survey*. DOI: 10.48550/ARXIV.1810.03548.

Wallelign, Serawork, Mihai Polceanu, and Cédric Buche (2018). "Soybean plant disease identification using convolutional neural network". In: *The thirty-first international flairs conference.*

Yu, Kun-Hsing, Andrew L Beam, and Isaac S Kohane (2018). "Artificial intelligence in healthcare". In: *Nature biomedical engineering* 2.10, pp. 719–731.

Yu, Xinjie et al. (2018). "Nondestructive freshness discriminating of shrimp using visible/near-infrared hyperspectral imaging technique and deep learning algorithm". In: *Food analytical methods* 11.3, pp. 768–780.

Coates, Dustin A. and Max Amordeluso (2019). *Voice applications for Alexa and Google assistant.* URL: https://www.amazon.com/alexa-voice-assistant/b?node=13727921011 (visited on 03/16/2023).

Garg, Adhesh et al. (2019). "Validation of random dataset using an efficient CNN model trained on MNIST handwritten dataset". In: *2019 6th International Conference on Signal Processing and Integrated Networks (SPIN).* IEEE, pp. 602–606.

Guo, Yang, Xuequn Shang, and Zhanhuai Li (2019). "Identification of cancer subtypes by integrating multiple types of transcriptomics data with deep learning in breast cancer". In: *Neurocomputing* 324, pp. 20–30.

Hang, Renlong et al. (2019). "Cascaded recurrent neural networks for hyperspectral image classification". In: *IEEE Transactions on Geoscience and Remote Sensing* 57.8, pp. 5384–5394.

He, Xin and Yushi Chen (2019). "Optimized input for CNN-based hyperspectral image classification using spatial transformer network". In: *IEEE Geoscience and Remote Sensing Letters* 16.12, pp. 1884–1888.

Jia, Wenyan et al. (2019). "Automatic food detection in egocentric images using artificial intelligence technology". In: *Public health nutrition* 22.7, pp. 1168–1179.

Liu, Hongyu et al. (2019). "CNN and RNN based payload classification methods for attack detection". In: *Knowledge-Based Systems* 163, pp. 332–341.

Panch, Trishan, Heather Mattie, and Leo Anthony Celi (2019). "The "inconvenient truth" about AI in healthcare". In: *NPJ digital medicine* 2.1, p. 77.

Selvaraj, Michael Gomez et al. (2019). "AI-powered banana diseases and pest detection". In: *Plant Methods* 15, pp. 1–11.

Shorten, Connor and Taghi M Khoshgoftaar (2019). "A survey on image data augmentation for deep learning". In: *Journal of big data* 6.1, pp. 1–48.

Tian, Zhi et al. (2019). "Fcos: Fully convolutional one-stage object detection". In: *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9627–9636.

Wang, Hai et al. (2019). "A comparative study of state-of-the-art deep learning algorithms for vehicle detection". In: *IEEE Intelligent Transportation Systems Magazine* 11.2, pp. 82–95.

Wang, Rui, Wei Li, and Liang Zhang (2019). "Blur image identification with ensemble convolution neural networks". In: *Signal Processing* 155, pp. 73–82.

Zhang, Han et al. (2019). "Self-attention generative adversarial networks". In: *International conference on machine learning*. PMLR, pp. 7354–7363.

Zhou, Lei et al. (2019). "Application of deep learning in food: A review". In: *Comprehensive Reviews in Food Science and Food Safety* 18.6, pp. 1793–1811.

Zhou, Xingyi, Dequan Wang, and Philipp Krähenbühl (2019). "Objects as points". In: *arXiv preprint arXiv:1904.07850*.

Liu, Zhining et al. (2020). "MESA: Boost Ensemble Imbalanced Learning with MEta-SAmpler". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., pp. 14463–14474.

Mohammed, Roweida, Jumanah Rawashdeh, and Malak Abdullah (2020). "Machine learning with oversampling and undersampling techniques: overview study and experimental results". In: *2020 11th international conference on information and communication systems (ICICS)*. IEEE, pp. 243–248.

Ozbay, Feyza Altunbey and Bilal Alatas (2020). "Fake news detection within online social media using supervised artificial intelligence algorithms". In: *Physica A: statistical mechanics and its applications* 540, p. 123174.

Rai, Arun (2020). "Explainable AI: From black box to glass box". In: *Journal of the Academy of Marketing Science* 48, pp. 137–141.

Shi, Haotian et al. (2020). "Automated heartbeat classification based on deep neural network with multiple input layers". In: *Knowledge-Based Systems* 188, p. 105036.

Singh, Richa, Sarthak Srivastava, and Rajan Mishra (2020). "Ai and iot based monitoring system for increasing the yield in crop production". In: *2020 International Conference on Electrical and Electronics Engineering (ICE3)*. IEEE, pp. 301–305.

Chamier, Lucas von et al. (2021). "Democratising deep learning for microscopy with ZeroCostDL4Mic". In: *Nature communications* 12.1, p. 2276.

Gc, S, Y Zhang, et al. (2021). *Using deep learning neural network in artificial intelligence technology to classify beef cuts. Front Sensors 2: 5.*

Guan, Hongjiao et al. (2021). "A Generalized Optimization Embedded Framework of Undersampling Ensembles for Imbalanced Classification". In: *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, pp. 1–10.

Han, Yifei et al. (2021). "Quality estimation of nuts using deep learning classification of hyperspectral imagery". In: *Computers and Electronics in Agriculture* 180, p. 105868.

Imblearn, Python (2021). *SMOTE*. Available at. URL: `https://imbalanced-learn.org/stable/references/generated/imblearn.over%5C_sampling.SMOTE.html` (visited on 05/10/2021).

McCarren, Andrew, Michael Scriney, et al. (Apr. 2021). *Meat Cut Image Dataset (BEEF)*. Zenodo. DOI: `10.5281/zenodo.4704391`. (Visited on 05/16/2022).

Neethirajan, Suresh and Bas Kemp (2021). "Digital twins in livestock farming". In: *Animals* 11.4, p. 1008.

Sampath, Vignesh et al. (2021). "A survey on generative adversarial networks for imbalance problems in computer vision tasks". In: *Journal of big Data* 8, pp. 1–59.

Susan, S. and A. Kumar (2021). "The balancing trick: Optimized sampling of imbalanced datasets:A brief survey of the recent State of the Art". In: *Engineering Reports* 3, p. 4.

Accounts, Irish Farm (2022). *Labour shortages in the Agri-Food Sector: Accounting amp; Financial Advisors.* URL: https://www.ifac.ie/news-insights/insights/labour-shortages-in-the-agri-food-sector (visited on 05/16/2023).

Agriculture, Department of (2022). *Assets.gov.ie.* URL: https://assets.gov.ie/238928/75008324-6f05-4f30-a9d3-6a9259371771.pdf (visited on 12/16/2022).

Ibec, MII (2022). *Mii factsheet employment in the meat sector - ibec.* URL: https://www.ibec.ie/-/media/documents/mii-documents/employment-in-the-meat-sector.pdf (visited on 05/16/2023).

Nayal, Kirti et al. (2022). "Exploring the role of artificial intelligence in managing agricultural supply chain risk to counter the impacts of the COVID-19 pandemic". In: *The International Journal of Logistics Management* 33.3, pp. 744–772.

Office, Central Statistics (Oct. 2022). *Meat Supply Balance 2021 - CSO - central statistics office.* URL: https://www.cso.ie/en/releasesandpublications/ep/p-msb/meatsupplybalance2021/ (visited on 02/16/2023).

Press, Gil (Oct. 2022). *A very short history of artificial intelligence (AI).* URL: https://www.forbes.com/sites/gilpress/2016/12/30/a-very-short-history-of-artificial-intelligence-ai/.

Zawish, Muhammad et al. (2022). "Toward On-Device AI and Blockchain for 6G-Enabled Agricultural Supply Chain Management". In: *IEEE Internet of Things Magazine* 5.2, pp. 160–166.

Group, Samsung (Mar. 2023). *Bixby: Apps amp; Services*. URL: `https://www.samsung.com/global/galaxy/apps/bixby/` (visited on 05/16/2023).

Jiang, C. et al. (2023). "Benchmarking state-of-the-art imbalanced data learning approaches for credit scoring". In: *Expert Systems with Applications* 213.

Office, Central Statistics (Apr. 2023). *Labour productivity - CSO - central statistics office*. URL: `https://www.cso.ie/en/releasesandpublications/ep/p-pii/productivityinireland2021/labourproductivity/` (visited on 05/10/2023).

Technology Lausanne, Swiss Federal Institute of (n.d.). *Food Image Dataset*. `https://www.epfl.ch/labs/mmspg/downloads/food-image-datasets/`, urldate = 2021-04-09.

Vivotek (n.d.). *IP 8362*. `https://www.vivotek.com/ip8362`, urldate = 2020-10-10.