# Computer Vision and Machine Learning Applications for Dairy Farming

by

Rafael Ehrich Pontes Ferreira

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Dairy Science)

at the

UNIVERSITY OF WISCONSIN-MADISON

2024

Date of final oral examination: 05/10/2024

The dissertation is approved by the following members of the Final Oral Committee:

João R. R. Dórea, Assistant Professor, Animal and Dairy Sciences
Guilherme J. M. Rosa, Professor, Animal and Dairy Sciences
Francisco Peñagaricano, Assistant Professor, Animal and Dairy Sciences
Kent A. Weigel, Professor, Animal and Dairy Sciences
Michael C. Ferris, Professor, Computer Sciences

# DEDICATION

*This dissertation is dedicated to my dad, Francisco, my mom, Andreia, my brother, Vinicius, my nephew, Felipe, and my niece, Beatriz*

# ACKNOWLEDGMENTS

First, I would like to thank my advisor and friend Dr. João Dórea for supporting me in every step of this journey. Since my very first days in Madison, you have always made me feel at home, even though I was thousands of miles away from my family in Brazil and did not know anyone else here. If I can now consider Madison my second home, it is largely due to your help in guiding me personally and professionally, and in being the best PhD advisor I could have asked for. Thank you for encouraging me to try all those seemingly crazy ideas that ended up actually working better than expected, and thank you for making all the work and effort that culminated in this dissertation much more enjoyable! You have become a role model for me in how to be a leader who gets things done while still being liked and admired by everyone. I also want to thank my committee members Dr. Guilherme Rosa, Dr. Francisco Peñagaricano, Dr. Kent Weigel, and Dr. Michael Ferris for all the orientation and guidance that you have provided throughout these years. I feel very privileged to have had the opportunity to work alongside such amazing researchers and to have learned so much from you.

Finishing a PhD is not easy, nor is it a one-person job. Thank you to all my labmates and colleagues who helped me in each research project and trial. I am especially grateful to Tiago Bresolin, Luiz Gustavo Pereira, Ariana Negreiro, and Guilherme Lobato for all the times we had to collect data at farms, install cameras and other systems, and figure out how to solve problems that gave everyone a terrible headache (whose idea was it to use small electronics in dairy farms?). Every challenge that we faced during those projects was made easier by your help and friendship, from having deep philosophical conversations to simply venting off about why the images stopped being collected or why the cows were not doing what we needed them to do.

Because my experience during those 5 years of graduate school involved much more than just study, research, and work, I want to thank all the friends that I've made during this time, who were essential for helping me go through these most intense and rewarding 5 years of my life. Among all the churrascos, parties, gatherings, summer trips, board game sessions, volleyball games, and nights spent awake talking about life, these were also the most fun 5 years that I have ever had! I won't mention any specific names this time because I know I will forget someone (I tend to do that) and I don't want people to be mad at me, but I am sure everyone who I am thinking about now knows that they are included in this. To make an exception, I want to thank Cora for being the best life partner I could have. Thank you for all the hours-long conversations, including patiently listening to me talk excitedly about cows, computers, and games. Thank you for sharing all the fun, exciting, difficult (but also rewarding), and ultimately memorable moments that we have been through in the last 4 years (and counting). There are still many more to come!

Finally, and most importantly, I want to thank all my family, especially my dad, Francisco, my mom, Andreia, my brother, Vinicius, my sister-in-law, Patricia, and their wonderful kids, Felipe and Bia. Thank you for helping shape who I am and my values, and for continuously teaching me what love is about. Thank you for making sure that I know there is a home that I will always love and where I will always be loved, no matter how far away I am physically. You make me feel like I never actually left home.

## ABSTRACT

With recent advancements in precision livestock farming (**PLF**) and machine learning (**ML**) techniques, computer vision systems (**CVS**) have gained popularity as powerful tools for individual animal monitoring. CVS can capture phenotypes from multiple individual animals at a time using a single device in an automated and non-intrusive manner. These systems require individual animal identification to match animals with their corresponding predicted phenotypes, which can be done via external identification systems, or using computer vision-based animal identification algorithms. Previous studies have proposed the use of computer vision techniques for individual identification of dairy cows based on their coat color pattern. However, these methods are limited to breeds that present such unique color patterns. Furthermore, no previous research has been done on the applicability of such methods in the long term, with animals experiencing visual changes due to body growth or different physiological states.

Chapter 1 introduces current applications of computer vision for animal identification, and, in Chapter 2, different methods are explored for using 3-dimensional representations of the dorsal surface of dairy calves to perform individual identification without relying on unique coat color patterns. Moreover, the proposed methods are evaluated on calves during their growth stage, assessing their performance as the animals experience changes in their body shape and size from weeks two to eight of life. The trained models achieved accuracies of up to 80.4% for identifying individual animals among 38 individuals using exclusively their 3D surface. Additionally, the evaluated algorithms were able to identify individuals among a group of five animals in their growing period with an accuracy of up to 85.6% even when skipping three weeks between the training and testing data.

Chapter 3 further explores the animal identification problem, evaluating the potential of a semi-supervised learning technique called pseudo-labeling for improving the accuracy of neural networks trained for animal identification. Modern computer vision algorithms such as convolutional neural networks (**CNN**) usually require large amounts of annotated data to train and generalize well to different environments. Semi-supervised learning techniques can leverage information contained in unlabeled datasets to improve the performance of trained models using smaller annotated datasets alongside larger amounts of unlabeled data. The results found were promising, showing that similar or even superior predictive performance was achieved using just a fraction of the annotated data when applying the proposed variation of pseudo-labeling. When using only 50% of the original labeled dataset, the final model resulting from pseudo-labeling achieved an accuracy of 89.7% for identifying individuals among 59 Holstein cows, exhibiting a significant improvement when compared to the 77.5% accuracy achieved when using the full dataset without performing pseudo-labeling. In addition, this technique is flexible enough to be applied to any previously trained image classification neural network, given that large unlabeled image datasets are available.

In Chapters 4 and 5, the focus is shifted to developing machine learning pipelines that integrate data from different domains for phenotype prediction, more specifically the early detection of postpartum subclinical ketosis (**SCK**) using exclusively prepartum data. In Chapter 4, multiple computer vision and image processing techniques are explored to extract features from depth images taken from a top-down view of the dorsal region of dairy cows. Natural language processing (**NLP**) and modern large language models (**LLM**) are leveraged to extract text embeddings from notes retrieved from farm management software. It was found that both image and text features contributed to improving the predictive performance of the trained ML models

when compared to using only tabular data containing behavior and cow history extracted from wearable sensors and the farm management software. Models incorporating image features achieved an average $F_1$ score of up to 0.706, while models incorporating text features achieved an average $F_1$ score of up to 0.681. These scores surpassed the average $F_1$ score of 0.655 achieved by ML models trained using only tabular data.

In Chapter 5, a cloud computing-based framework was proposed to automate the processing and integration of phenotypic and genotypic data. This framework integrates features extracted from genotype data, depth images, data collected from wearable sensors that monitor feeding behavior and activity, and historical data retrieved from the farm management software, and performs early detection of SCK using data fusion techniques and multimodal machine learning. The proposed pipeline follows a modularized approach where independent modules extract information from different types of data. From depth and infrared images, a cow body segmentation module removes all background pixels, an image quality assessment module removes from the pipeline images that do not conform to quality standards, an animal identification module identifies the animal present in the image, and a feature extraction module extracts body shape information from depth images using a trained neural network for body condition score (**BCS**) classification. From genotypic data, a feature extraction module automatically performs data cleaning and reduces the data dimensionality based on a reference genotype dataset. Finally, descriptive statistics are calculated from data collected from wearable sensors and farm management software and stored for future use. In this study, the features extracted from the different data modalities were used to perform early detection of SCK using exclusively prepartum data, but some of the implemented modules can be re-used for any phenotype prediction task that integrates body shape, genotype, and behavior information.

This dissertation highlights the potential of machine learning and computer vision in guiding data-driven management decisions in dairy farming, allowing for the implementation of practices that can improve farm profitability, productivity, and animal health. Animal identification is essential for individual animal monitoring, and CVS can leverage the images already being used for phenotype prediction to identify animals without the need for external devices. In addition, machine learning pipelines that integrate data from different sources can be implemented in cloud-computing platforms to automate and streamline the early detection of health issues during the transition period, which could support preventive actions in dairy farms, reducing costs associated with diseases, and improving animal health and welfare.

TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| 1D | 1-Dimensional |
| 2D | 2-Dimensional |
| 3D | 3-Dimensional |
| API | Application Programming Interface |
| BCS | Body Condition Score |
| BFT | Backfat Thickness |
| BHB | Beta-Hydroxybutyrate |
| BW | Body Weight |
| CDCB | Council on Dairy Cattle Breeding |
| CHTC | Center for High Throughput Computing |
| CNN | Convolutional Neural Network |
| CNN-RNN | Convolutional and Recurrent Neural Network |
| CORAL | Consistent Rank Logits |
| CORN | Conditional Ordinal Regression for Neural Networks |
| CSV | Comma-Separated Values |
| CVS | Computer Vision System |
| DCRC | Dairy Cattle Research Center |
| DIM | Days in Milk |
| DMI | Dry Matter Intake |
| DNN | Deep Neural Network |
| DRTC | Days Respective to Calving |
| FC | Fully-Connected |

| | |
|---|---|
| FPS | Frames Per Second |
| GPU | Graphics Processing Unit |
| HTTP | Hypertext Transfer Protocol |
| ID | Identification |
| IoU | Intersection over Union |
| LASSO | Least Absolute Shrinkage and Selection Operator |
| LLM | Large Language Models |
| LSTM | Long Short-Term Memory |
| MAD | Median Absolute Deviation |
| MAE | Mean Absolute Error |
| mAP | Mean Average Precision |
| MDI | Mean Decrease in Impurity |
| ML | Machine Learning |
| NEB | Negative Energy Balance |
| NEFA | Non-Esterified Fatty Acid |
| NLP | Natural Language Processing |
| OCR | Optical Character Recognition |
| OLS | Ordinal Least Squares |
| PCA | Principal Component Analysis |
| PCC | Pearson Correlation Coefficient |
| PLF | Precision Livestock Farming |
| PLS | Partial Least Squares |
| ReLU | Rectified Linear Unit |

| | |
|---|---|
| REST | Representational State Transfer |
| RF | Random Forest |
| RFID | Radio-Frequency Identification |
| RGB | Red, Green, Blue |
| RGB-D | Red, Green, Blue, Depth |
| RMSE | Root Mean Square Error |
| RNN | Recurrent Neural Network |
| SCK | Subclinical Ketosis |
| SD | Standard Deviation |
| SIFT | Scale-Invariant Feature Transform |
| SNP | Single-Nucleotide Polymorphism |
| SSL | Semi-Supervised Learning |
| SURF | Speeded-Up Robust Features |
| SVM | Support Vector Machine |
| UMAP | Uniform Manifold Approximation and Projection |

# CHAPTER ONE: LITERATURE REVIEW – COMPUTER VISION FOR ANIMAL IDENTIFICATION AND BODY CONDITION SCORING, AND DATA INTEGRATION VIA MULTIMODAL MACHINE LEARNING

## INTRODUCTION

In recent years, precision livestock farming (**PLF**) technologies have emerged as powerful tools for improving the efficiency and productivity of livestock farms. Such technologies empower more informed and rapid farm management decisions (Berckmans, 2017) and, by facilitating high-throughput phenotyping, can enhance the capabilities for genetic selection (Brito et al., 2020; Silva et al., 2021). Among PLF technologies, computer vision systems (**CVS**) have received great attention due to their potential for monitoring animals in a highly scalable and non-intrusive way, with few devices being able to collect phenotypes from multiple individuals at a time (Borges Oliveira et al., 2021). High-throughput phenotyping via CVS requires individual animal identification, which can be done either through external identification systems, or via computer vision algorithms using the same images collected for phenotyping (Hossain et al., 2022). CVS have been proposed to perform body condition score (**BCS**) evaluation (Qiao et al., 2021) for supporting successful transition period management strategies in dairy farms. Health problems generally depend on a multitude of factors affecting dairy cows, and the integration of data from different sources is imperative for developing robust machine learning models for individual animal health monitoring. In this context, multimodal machine learning (Baltrušaitis et al., 2019) and cloud computing (Schokker et al., 2022) can be powerful tools for supporting data-driven farm management decisions and improving animal health and welfare.

## COMPUTER VISION FOR ANIMAL IDENTIFICATION

Individual animal monitoring via CSV requires an automated way to identify the animals that are present in the collected images, which can be done through an external identification system such as radio-frequency identification (**RFID**) (Voulodimos et al., 2010), or via algorithms implemented in the CVS itself (Qiao et al., 2021; Hossain et al., 2022). Using computer vision techniques for both animal identification and phenotyping at the same time can prove beneficial by limiting the use of external devices and accessories attached to the animals, which reduce labor and time associated with manually installing and maintaining such devices (Adam et al., 2016), and animal welfare concerns (Johnston and Edwards, 1996; Chapa et al., 2020). Additionally, CVS for animal identification can be used as a tool for advancing traceability in the food supply chain, contributing to improved infectious diseases control, food safety, and consumer trust (Awad, 2016).

Previous studies proposing CVS for individual cattle identification can be divided according to three key characteristics: (1) the part of the animal used for identification, (2) the type of image or video captured and processed by the system, and (3) whether it was designed for closed-set or open-set identification. Existing approaches have explored the detection of unique visual features in the muzzle, retina, iris, face, and body of the animal, along with ear tag and collar digit reading, for cattle identification. Such approaches relied either on Red, Green, Blue (**RGB)**, Red, Green, Blue, Depth (**RGB-D**), depth images, or videos. Finally, most methods were designed for use with a fixed set of animals, requiring retraining or modifying the algorithms to include a new animal for identification, and some proposed techniques can perform open-set identification, being able to seamlessly include new individuals in the identification pipeline as new animals are added to the herd.

Previous works have explored the presence of unique animal biometrics for individual identification using CVS. Muzzle print images have been used for cattle identification by extracting unique features from the images using feature description and feature extraction methods such as Speeded-Up Robust Features (**SURF**) (Kumar et al., 2017), Local Binary Patterns (Kumar et al., 2017; Kusakunniran et al., 2018), Gabor filters (Tharwat et al., 2014; Kusakunniran et al., 2018), Principal Component Analysis (**PCA**) and Euclidean distance classifier (Barry et al., 2007), and deep neural networks (Kumar et al., 2018). Cattle muzzle patterns are known to be unique to each individual (Petersen, 1922), and methods that take advantage of this discovery tend to be very accurate (approximately 99% accuracy on datasets containing about 30 animals). However, it is often impractical to collect images from the muzzle of the animals in large scale commercial operations, as they require the animals to remain still and close to the cameras. Similarly, methods that take advantage of the iris and retinal biometric features (Allen et al., 2008; Sun et al., 2013a; Lu et al., 2014) achieve high accuracies of around 98% but it is often difficult to capture images of the retina or iris of live animals in commercial settings.

Inspired by human facial recognition and identification technologies, multiple approaches have been proposed to identify cattle using images of their whole face. Similarly to previous studies applied to muzzle, retina, and iris images, Cai and Li (2013), Kumar et al. (2016), and Kumar et al. (2017) performed feature engineering to extract texture-based features from the images using feature descriptors such as LBP, SURF and PCA, and achieved accuracies of between 92% and 95% on datasets containing 30, 120, and 500 animals, respectively. More recently, multiple studies have proposed the use of deep neural networks to automatically extract features from the images and perform classification. Yao et al. (2019), Yang et al. (2019), and Wang et al. (2020) used convolutional neural networks (**CNN**) and achieved accuracies between

93% and 95% on 200, 1,000, and 36 cows, respectively. Xu et al. (2022) and Li et al. (2022a) proposed the application of CNNs specifically designed for use in embedded systems and low latency settings and achieved accuracies of 91% and 98% on 90 and 103 cows, respectively. Bergamini et al. (2018) used multiple views of the face of each animal to train an embedding CNN that was optimized to find similar embeddings for images belonging to the same animal, or dissimilar embeddings otherwise. They then applied k-nearest neighbors on the embeddings to perform animal identification, achieving an accuracy of 82% on a testing set containing 52 cows. The accuracies reported in these studies were calculated in terms of the number of images correctly identified in testing datasets usually containing dozens of images per cow. However, the images in the testing sets were captured shortly after those in the training sets, often on the same day or only a few days apart. This introduces significant temporal biases into the analyses, and the long-term predictive capabilities of the proposed methods still need to be evaluated in future studies.

While it might be more manageable to capture images from the face of the animals than from their retina, iris, or muzzle, it is still notoriously difficult to collect good quality images of the animals facing the camera directly using automated image collection systems. In order to collect such images, the cameras must be positioned at a low enough height, which might allow the animals to reach them, or rely on the cows to look up at the camera when the images are being taken, which is unrealistic. Additionally, only a limited number of phenotypes can be visually assessed based on the cow's face, rendering such CVS ineffective when combined with high-throughput phenotyping algorithms. However, such approaches can still be helpful in situations when humans would handle the animals and are able to take pictures of them using portable systems, potentially improving animal traceability in certain scenarios.

Using side (Bhole et al., 2019; Shen et al., 2020) and multiple (Hu et al., 2020) views of the animal body, previous studies found accuracies of 97%, 97%, and 98% on datasets containing 105, 136, and 93 cows. The problem with using side view images of the animals is that such methods can only be applied in places where the cows walk by themselves in a fixed direction, without the possibility of other cows walking next to them and blocking the view. Even in such conditions, some occlusion may occur due to the fact the most corridors in dairy farms contain metal gates to restrict animal movement.

Using top-down view images from the dorsal region of Holstein cows, Andrew et al. (2016) used local feature descriptors, and Xiao et al. (2022) used shape and color features from binarized images to train Support Vector Machines (**SVM**) for cattle identification, achieving accuracies of 97% and 99% on groups of 40 and 48 cows, respectively. Zin et al. (2018) and Phyo et al. (2018) applied CNNs, which are more robust to lighting conditions and animal pose, and achieved accuracies of 97% and 96% on datasets containing 45 and 60 cows. Using sequences of top-down view images by combining a CNN for image feature extraction and a Recurrent Neural Network (**RNN**) for capturing temporal information, Andrew et al. (2017), Qiao et al. (2019), and Qiao et al. (2020) achieved accuracies of 98%, 91%, and 91% for identifying 23, 41, and 50 Holstein cows, finding better accuracies on their datasets than when using single-image approaches. Although the authors found better accuracies when performing image sequence classification instead of single image classification, the first might be more difficult to apply in farm settings where hardware and connectivity limitations make video capturing, processing, and storing prohibitive in large scale.

Methods that use top-down view images have some clear advantages when compared to other methods, as they can make use of cameras positioned close to the farm ceiling, out of reach from the animals, and with minimal possibilities for occlusion. In addition, proposed CVS can also

use top-down view images of the cows to predict other phenotypes, such as body weight and body condition score (Qiao et al., 2021), ribeye area and circularity (Caffarini et al., 2022), and behavior (Tsai and Huang, 2014). However, the previously discussed approaches rely on the existence of unique coat color patterns, which is only true for certain cattle breeds such as Holstein. Even when RGB-D images were used, the depth component was only used to facilitate cow body segmentation and the cattle identification was performed using exclusively the RGB channels.

Drawing from human gait identification studies, Okura et al. (2019) proposed extracting gait features from sequences of depth frames capturing cows in motion and found an accuracy of 76% on a testing dataset comprising 16 cows. Despite this modest accuracy compared to alternative methods, this approach's reliance solely on depth frames enables its application to any breed of cattle, rather than being restricted to those with distinct coat color patterns. However, it remains unclear whether this method's performance would be influenced by changes in the cows' gait over time. The evaluation sequences were obtained within three weeks of the training data, and the study did not investigate potential impacts from factors such as lameness or other conditions affecting gait.

Without relying on coat color patterns or depth images, Myat Noe et al. (2023) proposed a method that uses multi-object tracking algorithms to identify and track black cattle using RGB images. They achieved an accuracy of 97% in detecting and identifying 20 animals in a one-hour-long video captured at 25 frames per second (**FPS**). However, this method only works well on videos with a high frame rate, as it heavily relies on the animals not moving or changing pose too much between consecutive frames, since it uses the position and extracted features of bounding boxes of cows detected in previous frames to infer the identities of cows in current frames.

Shifting from identifying animal biometrics to performing optical character recognition (**OCR**) on identifying accessories (ear tags or collars) attached to the animals, Velez et al. (2013) achieved an accuracy of 90%, and Ilestrand (2017) achieved an accuracy of 98% on a manually curated image dataset for correctly recognizing digits. Bezen et al. (2020) proposed identifying animals based on a collar fitted around their neck and found an accuracy of 94% for correctly identifying the cows. Using videos instead of images, Zin et al. (2020a) and Smink et al. (2024) found accuracies of 84% and 71% for correctly identifying the cow identification number among 25 and 550 cows, respectively. Smink et al. (2024) found that only 79% of the ear tags were identifiable by human evaluators in any frame of the videos where they appeared, indicating that even accurate OCR techniques might struggle to correctly identify the animals based on images of their ear tags due to challenging conditions in commercial livestock farms. Ear tag and collar recognition approaches are interesting for large scale operations because no extra adjustment to the system should be required as new animals are introduced to the herd, and these approaches would work for any breed of animal regardless of whether they contain unique visual features or not, making it readily applicable to multiple farms and production systems. However, these approaches face similar challenges to face recognition methods regarding camera positioning and their synergy with other phenotyping algorithms. In addition, these methods still rely on proper management of ear tags or other accessories, which might be susceptible to human error, loss, or fraud, and raise animal welfare concerns (Johnston and Edwards, 1996).

Dairy farms are dynamic environments both in the sense that new animals are frequently introduced and removed from the herd, and that the animals themselves can change their visual characteristics significantly in the span of weeks depending on their growth stage, housing conditions, and management practice. Except for the approach proposed by Bergamini et al. (2018)

and the OCR-based methods, all the approaches previously mentioned were designed for use in a closed-set scenario, meaning that the systems were designed to identify a cow in an image among a fixed set of possible cows. This means that every time a new cow is included in the herd, the models need to be retrained or fine-tuned again, rendering the proposed methods unfeasible in large scale operations.

Bergamini et al. (2018) proposed a CNN-based approach that extracts embeddings from images of cow faces from multiple angles and validated their approach on an open-set setting, where some individuals in the testing set were not included in the training set, simulation real-world situation when a new cow is introduced to the system. However, the Top-1 accuracy found (accuracy for predicting the correct cow with the most confidence) among 52 individuals was 56%. Andrew et al. (2021) proposed a method based on deep metric learning using top-down view images of the dorsal region, on a dataset combining images from 46 cows taken indoors and outdoors. Using a combination of supervised softmax loss and reciprocal triplet loss to maximize embedding similarities between images of the same cow, and maximize dissimilarities for images of different cows, they found accuracies above 90% even when only 20% of the cows were included in the training set. Applying loss functions originally proposed for human facial recognition (Deng et al., 2018; Wang et al., 2018), Wang et al. (2024) found an accuracy of up to 95% on a testing set containing 8 cows, trained on a different set of 62. Combining the benefits of open-set animal identification and the use of depth images which potentially allows its application for any cattle breed, Sharma et al. (2024) trained deep metric learning networks based on ResNet-50 (He et al., 2016) using depth images and PointNet (Qi et al., 2016) using point clouds converted from the depth images. Similarly to Andrew et al. (2021), combinations of softmax and reciprocal or regular triplet loss (Schroff et al., 2015) were used, and they found an accuracy of 97% when

testing on a dataset containing 33 cows also included in the training set, and 66 cows that were not included in the training set.

The main difference between approaches designed for open-set and closed-set classification is that, while in closed-set there is a fixed number of classes (in this case, cows) that the images must belong to, in open-set scenarios each image is converted to a vector, also called embedding, that represents a form of signature of the cow present in the image. When new images are collected, their embeddings are extracted and compared to the embeddings of images for which the cow identification number is known (training set). If the embeddings are not close enough to any other embedding included in the training set, the image is considered as containing a new cow that has not been seen before by the model. In this case, a new cluster is formed in the embedding space, and future images can be classified as belonging to that same cow, and can be later assigned a certain cow identification number.

Although methods that perform animal identification based on ear tags, or retina would in theory be able to identify animals at any point in life since they are either independent of the animal itself, or use biometrics that do not change with time (Allen et al., 2008), other methods might be prone to errors if the characteristics that they rely on can change significantly as animals grow older or move to different locations. None of the previous studies have evaluated the performance of their proposed method in the medium or long term. Such methods were applied only to mature cows in a span of a few days at most, which is not enough time for significant changes to happen on their body, which could potentially affect the results. Many dairy farms introduce animals to their herd while they are still calves, meaning that their body will change considerably before they reach maturity. Thus, it is important to evaluate the performance of CVS for animal identification

in the medium and long term and explore how frequently new images of a calf need to be captured in order to maintain adequate prediction accuracy in the first months of life.

## COMPUTER VISION FOR BODY CONDITION SCORING

During the transition period, dairy cows can experience a negative energy balance (**NEB**) due to reduced feed intake and the high energy demands of early lactation (Grant and Albright, 1995; Drackley, 1999). Extensive research has been done to better understand the underlying metabolic and immune consequences of NEB and guide the development of management and nutritional practices aiming to overcome the increased risk of metabolic disorders that occur during that period (Grummer, 1995; Overton and Waldron, 2004; LeBlanc, 2010; Cardoso et al., 2020). The severity of NEB is associated with several health problems, such as hypocalcemia (Horst et al., 1994), retained placenta (Cameron et al., 1998), ketosis (Green et al., 1999), displaced abomasum (LeBlanc et al., 2005), metritis (Hammon et al., 2006), and endometritis (Dubuc et al., 2010). Such problems can cause great economic impact to a dairy farm due to reduced milk production and reproductive performance, high treatment costs, and increased culling rates (Steeneveld et al., 2020).

Cows with an elevated prepartum BCS have reduced fertility and milk production, and greater risks of developing metabolic disorders during early lactation due to an increased body fat mobilization (Buckley et al., 2003; Roche et al., 2009; Barletta et al., 2017; Daros et al., 2020). Because of that, monitoring prepartum BCS is crucial for a successful transition period management in dairy farms. However, assessing BCS in commercial farms is time-consuming, requires trained evaluators, and can lead to inconsistent results due to its subjective nature (Evans, 1978). CVS have been developed to perform BCS evaluation in a more automated and systematic way (Qiao et al., 2021). The approaches for automated BCS evaluation using image analysis and

computer vision can be split into methods that use 2-dimensional (**2D**) or 3-dimensional (**3D**) representations of the cow body shape.

BCS Prediction Using 2D Images

Among methods that proposed the use of 2D images for automated BCS assessment, Bewley et al. (2008), Battiato et al. (2010), and Azzaro et al. (2011) explored manually detecting 23 anatomical keypoints on the dorsal region of dairy cows and perform BCS prediction using shape descriptors based on those keypoints. They achieved accuracies of up to 93% within 0.25 deviation and mean absolute error (**MAE**) of 0.31. However, such methods require manual extraction of the keypoints, which might not be feasible in large scale real-time systems. This manual approach could be replaced by automated keypoint detection techniques such as the deep neural network proposed by Sun et al. (2013b), which could be trained using manually annotated images with the corresponding keypoints.

Halachmi et al. (2008) and Halachmi et al. (2013) proposed using the residuals of the cow contour on a fitted parabola to assess BCS based on how much the contour deviated from the parabola shape, finding a Pearson correlation coefficient (**PCC**) of 0.94 between predicted and manually assessed BCS. Nevertheless, their method was only evaluated on the training data itself, which might lead to overestimating results. An independent test set is necessary not only to more realistically validate the BCS prediction accuracy, but also to validate whether the automated frame selection pipeline required for the system to function properly performs adequately on images collected in different environments.

Bercovich et al. (2013) proposed automatically extracting 5 anatomical keypoints and a vector representing the tailhead contour from top-down view images of the tailhead area. Features such as distances and angles between the anatomical keypoints, as well as fourier descriptors of

the contour, were used for modeling BCS using linear regression and partial least squares (PLS) regression, finding accuracies of 50% and 100% within 0.25 and 0.75 deviation, respectively. The images utilized in this study were manually selected to guarantee that they conformed to the standards required by the system, which would not be feasible in commercial farms. Automated methods for detecting adequate images would be necessary, such as image classification CNNs (Li et al., 2022b).

Huang et al. (2019) and Li et al. (2019) used CNNs to automatically extract features from 2D RGB images of the tailhead area. Wu et al. (2021) compared multiple CNN architectures with Vision Transformers (Dosovitskiy et al., 2021) and Swin Transformers (Liu et al., 2021) for predicting BCS using RGB images and found that transformer architectures achieved superior performance on their dataset. These methods found impressive accuracies of up to 98% and 99% within 0.25 and 0.50 deviation. In those studies, however, the authors did not explicitly account for possible temporal biases in their datasets and performed a simple random split to determine training and testing sets. Deep neural networks such as CNNs and transformers are prone to overfit and perform exceptionally well on images that are too similar to the ones used for training, as they contain millions of parameters that can model the training data almost perfectly. Because of that, care should be taken to make sure that the trained models are tested on images taken from different cows, on different days, or even in different farms, to more realistically evaluate whether they can generalize to multiple environments.

BCS Prediction Using 3D Images

Salau et al. (2014) proposed a method for extracting 13 body shape traits from depth images based on 7 automatically found anatomical keypoints. The authors calculated the correlations between each trait and both BCS and backfat thickness (**BFT**), measured using ultrasound images,

but did not perform predictive modeling for BCS. The proposed method required manually selecting ideal images that had the anatomical keypoints successfully extracted from. Similarly to other methods, an automated way of selecting such images should be implemented for application in commercial settings.

Kuzuhara et al. (2015) calculated the geodesic distances between 6 pairs of manually detected anatomical keypoints located on the coccygeal ligament and on the hook, thurl, and pin bones, and trained linear regressions for predicting BCS, body weight, milk yield, milk fat, and milk protein, finding a coefficient of determination of up to 0.74 between predicted and observed BCS. Fischer et al. (2015) used PCA to project coordinates in normalized 3D surfaces to a common space with reduced dimensionality. The 3D surfaces were defined for each depth image based on 4 manually detected anatomical keypoints, and the trained model achieved root mean square error (**RMSE**) of 0.31 and PCC of 0.89 between predicted and observed BCS. Song et al. (2019) proposed a method for automatically identifying anatomical keypoints such as the vertebral column, sacral ligament, hook bone, and pin bone, and extracting features related to body shape using those keypoints. A trained model using such features achieved overall accuracy of 0.72 for predicting exact BCS values. However, their method required manually collecting images from the side and back of the cows, which might be challenging to do using automatic image capture systems. Liu et al. (2020) calculated 6 areas and volumes based on anatomical keypoints automatically detected using empirically defined parameters that worked well on their dataset, and found accuracies of 76% and 94% within 0.25 and 0.50 deviation. With all these approaches that rely on manually selecting ideal images, manually detecting anatomical keypoints, or empirically defining image processing parameters for finding the anatomical keypoints, methods for automatic

frame selection and generalized keypoint detection are imperative for application in large scale and generalization for diverse image datasets.

Spoliansky et al. (2016) extracted features from top-down view depth images of the cows' backs by finding the regions in the images with the highest correlations between their depth values and BCS. They achieved an accuracy of 91% when considering up to 0.50 deviation. Since the proposed method relies on extracting regions of 150 × 200 pixels from the original depth images of the cows and calculating features based on exact pixel locations, it is heavily dependent on image capture conditions such as image resolution and camera position.

Hansen et al. (2015) and Hansen et al. (2018) opted to utilize the rolling ball algorithm (Lee et al., 2005) instead of detecting predetermined anatomical keypoints. They applied this algorithm to calculate the angularity of the 3D surface obtained from a depth image of each cow's back, which was subsequently utilized in predicting BCS. An MAE of 0.21 was found in the training data, with 80% of the cows scored within 0.34 of the manual assessment. The authors found that their system seemingly presented less inconsistency than human evaluators when scoring the cows, but this conclusion was made based solely on manual visual inspection of the images. However, the authors also make an important point regarding the reliability of current manual BCS evaluation in dairy farms, emphasizing the inconsistencies found in human assessments. With current image analysis and computer vision techniques, this raises the question of whether the current standards for visually evaluating BCS are still the best way to quantify cow fat stores and body shape, as they introduce inconsistencies and human subjectivities into the assessment.

Similarly to Hansen et al. (2018), (Zin et al., 2020b) avoided the use of anatomical keypoints and extracted global roughness features from the 3D surfaces of the cows' backs. They

reported an MAE of 0.13 on an independent test set. However, all images used as a testing set exhibited BCS values between 3 and 3.75, providing no means to validate the efficacy of the proposed method in more extreme cases.

Rodríguez Alvarez et al. (2018) and Rodríguez Alvarez et al. (2019) trained CNNs using depth images with two additional channels: depth image filtered using Fourier transform to remove low spatial frequencies, and edges detected by the Canny algorithm (Canny, 1986) They achieved accuracies of 82% and 97% within 0.25 and 0.50 deviation, respectively. Similarly, Yukun et al. (2019) trained CNNs using images containing depth, gray, and phase congruency channels. Phase congruency was used instead of the previously explored Canny edge channel (Rodríguez Alvarez et al., 2018) because of the proximity between cows in each image, which caused the boundaries between cows to be too weak to be correctly extracted using depth images. Accuracies of 77% and 98% were found within 0.25 and 0.50 deviation. Zhao et al. (2023) trained CNNs on feature images constructed by calculating the vertical distances between each point in a voxelized 3D point cloud and the convex hull that surrounds it and achieved accuracies of 91% and 96% within 0.25 and 0.50 deviation. Finally, Shi et al. (2023) trained a neural network for automatically extracting features from 3D point clouds by adding an attention-based mechanism to the PointNet++ architecture (Qi et al., 2017), and found accuracies of 80% and 96% within 0.25 and 0.50 deviation, respectively.

Concluding Remarks

Although some of the proposed models have achieved great success in accurately predicting BCS based on 2D or 3D images, especially when considering 0.25 or 0.50 deviations, it is important to consider that BCS is ultimately a subjective measurement, and the human assessments used to train such models are prone to inconsistencies. Additionally, the quarter-point

divisions usually implemented for evaluating BCS do not account for subtle changes in body shape, or the distinction between different fat distribution profiles. Furthermore, the BCS variation of a cow through time can be more important than absolute BCS values for health and reproductive performance in some cases (Barletta et al., 2017). Considering the current scenario of advanced computer vision and data analysis, it might be beneficial to define other more quantitative standards of cow fat stores based on measures of shape, depth, form, and contour of different regions of the cow's body. BCS is primarily used to assess metabolic disorder risks, productive and reproductive performance, fertility, and overall cow health, while also facilitating dietary adjustments. Thus, the precision livestock community could progress towards using cow quantitative information extracted from computer vision systems (for example, using feature extraction methods proposed in previous works, or features extracted from deep neural networks) to directly predict such cow performance and health metrics and assess risks of disease and other production metrics. Ideally, new gold standards might arise which are less prone to human error and subjectivity. Finally, it is necessary to collect a significant number of examples of cows in both extremes of body condition in order to train models with a good representation of the possible states that a cow's body could be in different stages of their life. This is relevant both when the desired predicted variable is BCS itself or any other representation of body shape.

## MULTIMODAL MACHINE LEARNING

Health problems in dairy cows are complex and can be influenced by a multitude of factors, making diagnosis and management challenging. Not only current fat stores and fat mobilization can indicate the risk of a cow developing metabolic disorders, but also feeding behavior, activity levels, diet composition, current physiological state, genetic predisposition, environmental conditions, and management practices can all impact on a dairy cow's health (Ingvartsen and

Moyes, 2013; Overton et al., 2017). PLF technologies generate data for tracking such variables, which can be used to develop machine learning models for monitoring animal health (Berckmans, 2017; García et al., 2020). In general, individual animal data is available in the form of data tables, images, or text. When using high-dimensional or unstructured data such as text, images, or genetic datasets, features must be extracted, and the data dimensionality should be reduced before adequate use in machine learning models. High-dimensional data cause predictive analyses to suffer from the so-called curse of dimensionality, which is the phenomenon that predictive models tend to overfit and not perform well with a growing number of input dimensions (Köppen, 2000). Many feature extraction and dimensionality reduction techniques exist to help solve this problem by projecting high-dimensional data into lower-dimensional representations (Jia et al., 2022).

When using genomic data for phenotype prediction, feature extraction and selection can be performed via traditional statistical analysis (Manthena et al., 2022), machine learning methods (Feldner-Busztin et al., 2023), or, more recently, deep neural networks (Eraslan et al., 2019; Zou et al., 2019). In genomics, feature extraction and selection are useful not only for reducing computational requirements while maintaining acceptable predictive performance, but also for facilitating the understanding of the underlying biological factors that are relevant to the phenotype of interest. In addition, dimensionality reduction techniques generally improve the performance of predictive models due to the aforementioned curse of dimensionality.

From image data, features can be extracted using traditional image processing and computer vision techniques, or deep learning methods (O'Mahony et al., 2020). Techniques such as SURF (Bay et al., 2006) and Scale-Invariant Feature Transform (**SIFT**) (Karami et al., 2017) can provide descriptors of automatically detected points of interest in an image. These descriptors compute information related to texture, color gradients, and other local features, which can be used

as inputs to machine learning algorithms such as linear regressions and Support Vector Machines (Hearst et al., 1998) to perform classification or identification of objects in a scene based on training or reference images. Other techniques such as Hough transforms (Goldenshluger and Zeevi, 2004), Gabor filters, or histogram of oriented gradients (**HOG**) (Freeman and Roth, 1995) can extract features related to the presence of arbitrary shapes and textures, which can similarly be used to perform image classification using machine learning algorithms.

More recently, CNNs gained popularity in computer vision tasks such as image classification, instance segmentation, and object detection (Li et al., 2022b). These neural networks can automatically learn optimal filters based on the training data, detecting patterns in the images that are most important for a given task. The convolutional layers that form the core of CNNs extract features from images by applying automatically learned filters through a convolution operation, removing the need for manually engineering filters such as Gabor or Haar filters (Haselhoff and Kummert, 2009). The main drawback of CNNs is that they usually require large amounts of annotated data to be able to automatically learn such convolutional filters, which can be time-consuming and expensive to acquire. Techniques such as transfer learning (Weiss et al., 2016), semi-supervised (Oliver et al., 2018), and self-supervised learning (Jing and Tian, 2021) have been proposed to reduce the need for large scale annotated data and enable CNNs to be trained using smaller datasets.

Drawing inspiration from their success in natural language processing (**NLP**), special neural networks called Transformers have been adapted to computer vision tasks via Vision Transformers (**ViT**) (Dosovitskiy et al., 2021), introducing the concept of patch embeddings to interpret images as sequences of patches similarly to how text is interpreted as a sequence of words. Although deep neural networks based on the ViT architecture can achieve superior results in

certain scenarios when compared to CNNs, they generally require even larger amounts of data to be trained, as they lack some of the inductive biases present in CNNs, such as translation invariance, pixel locality, and two-dimensional neighborhood structure. These inductive biases make CNNs more efficient but less flexible to learn image features for downstream tasks, so when sufficiently large datasets are available, ViT can perform better (Dosovitskiy et al., 2021).

In the pursuit of constructing general-purpose neural networks for image understanding, the concept of foundation models has gained popularity in recent years (Bommasani et al., 2021). Such models are either trained on very large, annotated datasets spanning a multitude of contexts and domains, or on even larger unlabeled datasets using self-supervised learning techniques (Jing and Tian, 2021). These models typically contain a larger number of parameters, enabling them to learn to extract features from different types of images for various tasks, rather than being optimized for a specific application. This allows the same foundation model to be used to extract features from images for several applications at the same time, without the need for re-training or fine-tuning it.

In the field of natural language processing, autoregressive techniques such as word2vec (Mikolov et al., 2013) have been proposed for converting words into a vector space that encapsulates their semantic and syntactic information. With the advancement of deep learning, RNNs have been proposed to automatically learn to extract embeddings (vectors) from words or tokens (parts of words) based on the training data and process them through recurrent layers to perform text classification or generation (Liu et al., 2016). Using attention mechanisms to identify the most important parts of the text and how different words are interconnected, a neural network architecture called Transformers (Vaswani et al., 2017) achieved great success in NLP tasks. Similarly to RNNs, these networks can learn to extract embeddings from words and sentences,

synthesizing their semantics with respect to the context within the input text. This allows such models to extract text embeddings that contain useful information about the text, which can be used as features for different classification and generation tasks without the need for re-training or fine-tuning. These models are usually trained using a large corpus of text to perform next- or masked-word prediction following a self-supervision paradigm, in which case they are generally called large language models (**LLM**). When trained this way, these models can learn structural information about a language, and the meaning of words and how they interact with each other within the context of the input text.

After features are extracted from each type of data (also called data modality), they can be combined for use in predictive modeling using techniques that can be categorized as early, late, or hybrid fusion. Early fusion consists of combining the features from different modalities before training the machine learning (**ML**) models, late fusion consists of training separate ML models for each modality and combining their predictions into a final prediction possibly through another ML model, and hybrid fusion is a combination of early and late fusion, borrowing mechanisms from both (Lahat et al., 2015). Within hybrid fusion, methods based on machine learning and deep learning algorithms have been proposed (Gao et al., 2020; Meng et al., 2020), and it is currently an active area of research.

After ML models are trained, they can be deployed either locally at the farm, via edge computing, or completely on the cloud. Systems deployed locally have the advantage of not requiring internet connectivity to function, a benefit particularly relevant in areas with limited internet access, such as rural areas. However, since the models run on devices located at the farm often with no internet access, data integration with other systems outside of the farm premises is limited, and scaling the system is expensive, as it requires replacing or including more hardware

physically. Edge computing extends local deployment by distributing computation to edge devices closer to where the data is generated, reducing latency and bandwidth requirements (Alonso et al., 2020). However, edge deployment may face challenges related to managing distributed systems, ensuring consistent model updates across devices, and scaling hardware capabilities. Conversely, cloud computing centralizes computation and storage resources in remote data centers, offering virtually unlimited scalability and flexibility, and improving data availability (Schokker et al., 2022). It facilitates rapid deployment and updates of machine learning models, enabling efficient resource utilization and cost-effectiveness. Nevertheless, reliance on internet connectivity can limit its application on farms.

In summary, multimodal machine learning combined with edge and cloud computing technologies can support the use of PLF applications for improving management decisions in dairy farms. ML models that utilize various dimensions of an individual animal with data originating from different sources can be constructed for robust prediction of health issues, productivity, and reproductive performance. The implementation of integrated PLF systems guides a data-driven approach to livestock farming, facilitating cost reduction, productivity enhancement, and advancements in animal health and welfare.

## REFERENCES

Adam, B.D., R. Holcomb, M. Buser, B. Mayfield, J. Thomas, C.A. O'Bryan, P. Crandall, D. Knipe, R. Knipe, and S.C. Ricke. 2016. Enhancing Food Safety, Product Quality, and Value-Added in Food Supply Chains Using Whole-Chain Traceability. International Food and Agribusiness Management Review 24. doi:https://doi.org/10.22004/ag.econ.240706.

Allen, A., B. Golden, M. Taylor, D. Patterson, D. Henriksen, and R. Skuce. 2008. Evaluation of retinal imaging technology for the biometric identification of bovine animals in Northern Ireland. Livest Sci 116:42–52. doi:https://doi.org/10.1016/j.livsci.2007.08.018.

Alonso, R.S., I. Sittón-Candanedo, Ó. García, J. Prieto, and S. Rodríguez-González. 2020. An intelligent Edge-IoT platform for monitoring livestock and crops in a dairy farming scenario. Ad Hoc Networks 98:102047. doi:https://doi.org/10.1016/j.adhoc.2019.102047.

Andrew, W., J. Gao, S. Mullan, N. Campbell, A.W. Dowsey, and T. Burghardt. 2021. Visual identification of individual Holstein-Friesian cattle via deep metric learning. Comput Electron Agric 185:106133. doi:https://doi.org/10.1016/j.compag.2021.106133.

Andrew, W., C. Greatwood, and T. Burghardt. 2017. Visual Localisation and Individual Identification of Holstein Friesian Cattle via Deep Learning. Pages 2850–2859 in 2017 IEEE International Conference on Computer Vision Workshops (ICCVW).

Andrew, W., S. Hannuna, N. Campbell, and T. Burghardt. 2016. Automatic individual holstein friesian cattle identification via selective local coat pattern matching in RGB-D imagery. Pages 484–488 in 2016 IEEE International Conference on Image Processing (ICIP).

Awad, A.I. 2016. From classical methods to animal biometrics: A review on cattle identification and tracking. Comput Electron Agric 123:423–435. doi:https://doi.org/10.1016/j.compag.2016.03.014.

Azzaro, G., M. Caccamo, J.D. Ferguson, S. Battiato, G.M. Farinella, G.C. Guarnera, G. Puglisi, R. Petriglieri, and G. Licitra. 2011. Objective estimation of body condition score by modeling cow body shape from digital images. J Dairy Sci 94:2126–2137. doi:https://doi.org/10.3168/jds.2010-3467.

Baltrušaitis, T., C. Ahuja, and L.-P. Morency. 2019. Multimodal Machine Learning: A Survey and Taxonomy. IEEE Trans Pattern Anal Mach Intell 41:423–443. doi:10.1109/TPAMI.2018.2798607.

Barletta, R. V, M. Maturana Filho, P.D. Carvalho, T.A. Del Valle, A.S. Netto, F.P. Rennó, R.D. Mingoti, J.R. Gandra, G.B. Mourão, P.M. Fricke, R. Sartori, E.H. Madureira, and M.C. Wiltbank. 2017. Association of changes among body condition score during the transition period with NEFA and BHBA concentrations, milk production, fertility, and health of Holstein cows. Theriogenology 104:30–36. doi:https://doi.org/10.1016/j.theriogenology.2017.07.030.

Barry, B., U. A. Gonzales-Barron, K. McDonnell, F. Butler, and S. Ward. 2007. Using Muzzle Pattern Recognition as a Biometric Approach for Cattle Identification. Trans ASABE 50:1073–1080. doi:https://doi.org/10.13031/2013.23121.

Battiato, S., G.M. Farinella, G.C. Guarnera, G. Puglisi, G. Azzaro, and M. Caccamo. 2010. Assessment of Cow's Body Condition Score Through Statistical Shape Analysis and Regression Machines. Pages 66–73 in Proceedings of the First Workshop on Applications of Pattern Analysis. PMLR, Cumberland Lodge, Windsor, UK.

Bay, H., T. Tuytelaars, and L. Van Gool. 2006. SURF: Speeded Up Robust Features. Pages 404–417 in Computer Vision – ECCV 2006. Springer Berlin Heidelberg, Berlin, Heidelberg.

Berckmans, D. 2017. General introduction to precision livestock farming. Animal Frontiers 7:6–11. doi:10.2527/af.2017.0102.

Bercovich, A., Y. Edan, V. Alchanatis, U. Moallem, Y. Parmet, H. Honig, E. Maltz, A. Antler, and I. Halachmi. 2013. Development of an automatic cow body condition scoring using body shape signature and Fourier descriptors. J Dairy Sci 96:8047–8059. doi:https://doi.org/10.3168/jds.2013-6568.

Bergamini, L., A. Porrello, A.C. Dondona, E. Del Negro, M. Mattioli, N. D'alterio, and S. Calderara. 2018. Multi-views Embedding for Cattle Re-identification. Pages 184–191 in 2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS).

Bewley, J.M., A.M. Peacock, O. Lewis, R.E. Boyce, D.J. Roberts, M.P. Coffey, S.J. Kenyon, and M.M. Schutz. 2008. Potential for Estimation of Body Condition Scores in Dairy Cattle from Digital Images. J Dairy Sci 91:3439–3453. doi:https://doi.org/10.3168/jds.2007-0836.

Bezen, R., Y. Edan, and I. Halachmi. 2020. Computer vision system for measuring individual cow feed intake using RGB-D camera and deep learning algorithms. Comput Electron Agric 172:105345. doi:https://doi.org/10.1016/j.compag.2020.105345.

Bhole, A., O. Falzon, M. Biehl, and G. Azzopardi. 2019. A Computer Vision Pipeline that Uses Thermal and RGB Images for the Recognition of Holstein Cattle. Pages 108–119 in Computer Analysis of Images and Patterns. Springer International Publishing, Cham.

Bommasani, R., D.A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M.S. Bernstein, J. Bohg, A. Bosselut, and E. Brunskill. 2021. On the opportunities and risks of foundation models. arXiv preprint arXiv:2108.07258.

Borges Oliveira, D.A., L.G. Ribeiro Pereira, T. Bresolin, R.E. Pontes Ferreira, and J.R. Reboucas Dorea. 2021. A review of deep learning algorithms for computer vision systems in livestock. Livest Sci 253:104700. doi:https://doi.org/10.1016/j.livsci.2021.104700.

Brito, L.F., H.R. Oliveira, B.R. McConn, A.P. Schinckel, A. Arrazola, J.N. Marchant-Forde, and J.S. Johnson. 2020. Large-Scale Phenotyping of Livestock Welfare in Commercial Production Systems: A New Frontier in Animal Breeding. Front Genet 11.

Buckley, F., K. O'Sullivan, J.F. Mee, R.D. Evans, and P. Dillon. 2003. Relationships Among Milk Yield, Body Condition, Cow Weight, and Reproduction in Spring-Calved Holstein-Friesians. J Dairy Sci 86:2308–2319. doi:https://doi.org/10.3168/jds.S0022-0302(03)73823-5.

Caffarini, J.G., T. Bresolin, and J.R.R. Dorea. 2022. Predicting ribeye area and circularity in live calves through 3D image analyses of body surface. J Anim Sci 100:skac242. doi:10.1093/jas/skac242.

Cai, C., and J. Li. 2013. Cattle face recognition using local binary pattern descriptor. Pages 1–4 in 2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference.

Cameron, R.E.B., P.B. Dyk, T.H. Herdt, J.B. Kaneene, R. Miller, H.F. Bucholtz, J.S. Liesman, M.J. Vandehaar, and R.S. Emery. 1998. Dry Cow Diet, Management, and Energy Balance as Risk Factors for Displaced Abomasum in High Producing Dairy Herds. J Dairy Sci 81:132–139. doi:https://doi.org/10.3168/jds.S0022-0302(98)75560-2.

Canny, J. 1986. A Computational Approach to Edge Detection. IEEE Trans Pattern Anal Mach Intell PAMI-8:679–698. doi:10.1109/TPAMI.1986.4767851.

Cardoso, F.C., K.F. Kalscheur, and J.K. Drackley. 2020. Symposium review: Nutrition strategies for improved health, production, and fertility during the transition period. J Dairy Sci 103:5684–5693. doi:https://doi.org/10.3168/jds.2019-17271.

Chapa, J.M., K. Maschat, M. Iwersen, J. Baumgartner, and M. Drillich. 2020. Accelerometer systems as tools for health and welfare assessment in cattle and pigs – A review. Behavioural Processes 181:104262. doi:https://doi.org/10.1016/j.beproc.2020.104262.

Daros, R.R., H.K. Eriksson, D.M. Weary, and M.A.G. von Keyserlingk. 2020. The relationship between transition period diseases and lameness, feeding time, and body condition during the dry period. J Dairy Sci 103:649–665. doi:https://doi.org/10.3168/jds.2019-16975.

Deng, J., J. Guo, and S. Zafeiriou. 2018. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. CoRR abs/1801.07698.

Dosovitskiy, A., L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.

Drackley, J.K. 1999. Biology of Dairy Cows During the Transition Period: the Final Frontier?. J Dairy Sci 82:2259–2273. doi:https://doi.org/10.3168/jds.S0022-0302(99)75474-3.

Dubuc, J., T.F. Duffield, K.E. Leslie, J.S. Walton, and S.J. LeBlanc. 2010. Definitions and diagnosis of postpartum endometritis in dairy cows. J Dairy Sci 93:5225–5233. doi:https://doi.org/10.3168/jds.2010-3428.

Eraslan, G., Ž. Avsec, J. Gagneur, and F.J. Theis. 2019. Deep learning: new computational modelling techniques for genomics. Nat Rev Genet 20:389–403. doi:10.1038/s41576-019-0122-6.

Evans, D.G. 1978. The interpretation and analysis of subjective body condition scores. Animal Science 26:119–125. doi:DOI: 10.1017/S0003356100039520.

Feldner-Busztin, D., P. Firbas Nisantzis, S.J. Edmunds, G. Boza, F. Racimo, S. Gopalakrishnan, M.T. Limborg, L. Lahti, and G.G. de Polavieja. 2023. Dealing with dimensionality: the application of machine learning to multi-omics data. Bioinformatics 39:btad021. doi:10.1093/bioinformatics/btad021.

Fischer, A., T. Luginbühl, L. Delattre, J.M. Delouard, and P. Faverdin. 2015. Rear shape in 3 dimensions summarized by principal component analysis is a good predictor of body condition score in Holstein dairy cows. J Dairy Sci 98:4465–4476. doi:https://doi.org/10.3168/jds.2014-8969.

Freeman, W.T., and M. Roth. 1995. Orientation histograms for hand gesture recognition. Pages 296–301 in International workshop on automatic face and gesture recognition. Citeseer.

Gao, J., P. Li, Z. Chen, and J. Zhang. 2020. A Survey on Deep Learning for Multimodal Data Fusion. Neural Comput 32:829–864. doi:10.1162/neco_a_01273.

García, R., J. Aguilar, M. Toro, A. Pinto, and P. Rodríguez. 2020. A systematic literature review on the use of machine learning in precision livestock farming. Comput Electron Agric 179:105826. doi:https://doi.org/10.1016/j.compag.2020.105826.

Goldenshluger, A., and A. Zeevi. 2004. The Hough transform estimator. The Annals of Statistics 32:1908–1932. doi:10.1214/009053604000000760.

Grant, R.J., and J.L. Albright. 1995. Feeding behavior and management factors during the transition period in dairy cattle. J Anim Sci 73:2791–2803. doi:10.2527/1995.7392791x.

Green, B.L., B.W. McBride, D. Sandals, K.E. Leslie, R. Bagg, and P. Dick. 1999. The Impact of a Monensin Controlled-Release Capsule on Subclinical Ketosis in the Transition Dairy Cow. J Dairy Sci 82:333–342. doi:https://doi.org/10.3168/jds.S0022-0302(99)75240-9.

Grummer, R.R. 1995. Impact of changes in organic nutrient metabolism on feeding the transition dairy cow. J Anim Sci 73:2820–2833. doi:10.2527/1995.7392820x.

Halachmi, I., M. Klopčič, P. Polak, D.J. Roberts, and J.M. Bewley. 2013. Automatic assessment of dairy cattle body condition score using thermal imaging. Comput Electron Agric 99:35–40. doi:https://doi.org/10.1016/j.compag.2013.08.012.

Halachmi, I., P. Polak, D.J. Roberts, and M. Klopcic. 2008. Cow Body Shape and Automation of Condition Scoring. J Dairy Sci 91:4444–4451. doi:https://doi.org/10.3168/jds.2007-0785.

Hammon, D.S., I.M. Evjen, T.R. Dhiman, J.P. Goff, and J.L. Walters. 2006. Neutrophil function and energy status in Holstein cows with uterine health disorders. Vet Immunol Immunopathol 113:21–29. doi:https://doi.org/10.1016/j.vetimm.2006.03.022.

Hansen, M., M. Smith, L. Smith, I. Hales, and D. Forbes. 2015. Non-intrusive automated measurement of dairy cow body condition using 3D video. Proceedings of the Machine Vision of Animals and their Behaviour (MVAB) 1.

Hansen, M.F., M.L. Smith, L.N. Smith, K. Abdul Jabbar, and D. Forbes. 2018. Automated monitoring of dairy cow body condition, mobility and weight using a single 3D video capture device. Comput Ind 98:14–22. doi:https://doi.org/10.1016/j.compind.2018.02.011.

Haselhoff, A., and A. Kummert. 2009. A vehicle detection system based on Haar and Triangle features. Pages 261–266 in 2009 IEEE Intelligent Vehicles Symposium.

He, K., X. Zhang, S. Ren, and J. Sun. 2016. Deep residual learning for image recognition. Pages 770–778 in Proceedings of the IEEE conference on computer vision and pattern recognition.

Hearst, M.A., S.T. Dumais, E. Osuna, J. Platt, and B. Scholkopf. 1998. Support vector machines. IEEE Intelligent Systems and their Applications 13:18–28. doi:10.1109/5254.708428.

Horst, R.L., J.P. Goff, and T.A. Reinhardt. 1994. Calcium and Vitamin D Metabolism in the Dairy Cow. J Dairy Sci 77:1936–1951. doi:https://doi.org/10.3168/jds.S0022-0302(94)77140-X.

Hossain, M.E., M.A. Kabir, L. Zheng, D.L. Swain, S. McGrath, and J. Medway. 2022. A systematic review of machine learning techniques for cattle identification: Datasets, methods and future directions. Artificial Intelligence in Agriculture 6:138–155. doi:https://doi.org/10.1016/j.aiia.2022.09.002.

Hu, H., B. Dai, W. Shen, X. Wei, J. Sun, R. Li, and Y. Zhang. 2020. Cow identification based on fusion of deep parts features. Biosyst Eng 192:245–256. doi:https://doi.org/10.1016/j.biosystemseng.2020.02.001.

Huang, X., Z. Hu, X. Wang, X. Yang, J. Zhang, and D. Shi. 2019. An Improved Single Shot Multibox Detector Method Applied in Body Condition Score for Dairy Cows. Animals 9. doi:10.3390/ani9070470.

Ilestrand, M. 2017. Automatic Eartag Recognition on Dairy Cows in Real Barn Environment. Linköping University, Computer Vision,.

Ingvartsen, K.L., and K. Moyes. 2013. Nutrition, immune function and health of dairy cattle. Animal 7:112–122. doi:10.1017/S175173111200170X.

Jia, W., M. Sun, J. Lian, and S. Hou. 2022. Feature dimensionality reduction: a review. Complex & Intelligent Systems 8:2663–2693. doi:10.1007/s40747-021-00637-x.

Jing, L., and Y. Tian. 2021. Self-Supervised Visual Feature Learning With Deep Neural Networks: A Survey. IEEE Trans Pattern Anal Mach Intell 43:4037–4058. doi:10.1109/TPAMI.2020.2992393.

Johnston, A.M., and D.S. Edwards. 1996. Welfare implications of identification of cattle by ear tags. Veterinary Record 138:612–614. doi:https://doi.org/10.1136/vr.138.25.612.

Karami, E., M. Shehata, and A. Smith. 2017. Image identification using SIFT algorithm: performance analysis against different image deformations. arXiv preprint arXiv:1710.02728.

Köppen, M. 2000. The curse of dimensionality. Pages 4–8 in 5th online world conference on soft computing in industrial applications (WSC5).

Kumar, S., A. Pandey, K. Sai Ram Satwik, S. Kumar, S.K. Singh, A.K. Singh, and A. Mohan. 2018. Deep learning framework for recognition of cattle using muzzle point image pattern. Measurement 116:1–17. doi:https://doi.org/10.1016/j.measurement.2017.10.064.

Kumar, S., S.K. Singh, and A.K. Singh. 2017a. Muzzle point pattern based techniques for individual cattle identification. IET Image Process 11:805–814. doi:https://doi.org/10.1049/iet-ipr.2016.0799.

Kumar, S., S.K. Singh, R. Singh, and A.K. Singh. 2017b. Recognition of Cattle Using Face Images. S. Kumar, S.K. Singh, R. Singh, and A.K. Singh, ed. Springer Singapore, Singapore.

Kumar, S., S. Tiwari, and S.K. Singh. 2016. Face recognition for cattle. Pages 65–72 in Proceedings of 2015 3rd International Conference on Image Information Processing, ICIIP 2015.

Kusakunniran, W., A. Wiratsudakul, U. Chuachan, S. Kanchanapreechakorn, and T. Imaromkul. 2018. Automatic cattle identification based on fusion of texture features extracted from

muzzle images. Pages 1484–1489 in 2018 IEEE International Conference on Industrial Technology (ICIT).

Kuzuhara, Y., K. Kawamura, R. Yoshitoshi, T. Tamaki, S. Sugai, M. Ikegami, Y. Kurokawa, T. Obitsu, M. Okita, T. Sugino, and T. Yasuda. 2015. A preliminarily study for predicting body weight and milk properties in lactating Holstein cows using a three-dimensional camera system. Comput Electron Agric 111:186–193. doi:https://doi.org/10.1016/j.compag.2014.12.020.

Lahat, D., T. Adali, and C. Jutten. 2015. Multimodal Data Fusion: An Overview of Methods, Challenges, and Prospects. Proceedings of the IEEE 103:1449–1477. doi:10.1109/JPROC.2015.2460697.

LeBlanc, S. 2010. Monitoring Metabolic Health of Dairy Cattle in the Transition Period. Journal of Reproduction and Development 56:S29–S35. doi:10.1262/jrd.1056S29.

LeBlanc, S.J., K.E. Leslie, and T.F. Duffield. 2005. Metabolic Predictors of Displaced Abomasum in Dairy Cattle. J Dairy Sci 88:159–170. doi:https://doi.org/10.3168/jds.S0022-0302(05)72674-6.

Lee, J.R.J., M.L. Smith, L.N. Smith, and P.S. Midha. 2005. A mathematical morphology approach to image based 3D particle shape analysis. Mach Vis Appl 16:282–288. doi:10.1007/s00138-005-0181-x.

Li, X., Z. Hu, X. Huang, T. Feng, X. Yang, and M. Li. 2019. Cow Body Condition Score Estimation with Convolutional Neural Networks. Pages 433–437 in 2019 IEEE 4th International Conference on Image, Vision and Computing (ICIVC).

Li, Z., X. Lei, and S. Liu. 2022a. A lightweight deep learning model for cattle face recognition. Comput Electron Agric 195:106848. doi:https://doi.org/10.1016/j.compag.2022.106848.

Li, Z., F. Liu, W. Yang, S. Peng, and J. Zhou. 2022b. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. IEEE Trans Neural Netw Learn Syst 33:6999–7019. doi:10.1109/TNNLS.2021.3084827.

Liu, D., D. He, and T. Norton. 2020. Automatic estimation of dairy cattle body condition score from depth image using ensemble model. Biosyst Eng 194:16–27. doi:https://doi.org/10.1016/j.biosystemseng.2020.03.011.

Liu, P., X. Qiu, and X. Huang. 2016. Recurrent neural network for text classification with multi-task learning. arXiv preprint arXiv:1605.05101.

Liu, Z., Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. 2021. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. Pages 10012–10022 in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV).

Lu, Y., X. He, Y. Wen, and P.S.P. Wang. 2014. A new cow identification system based on iris analysis and recognition. Int J Biom 6:18–32. doi:https://dx.doi.org/10.1504/IJBM.2014.059639.

Manthena, V., D. Jarquín, R.K. Varshney, M. Roorkiwal, G.P. Dixit, C. Bharadwaj, and R. Howard. 2022. Evaluating dimensionality reduction for genomic prediction. Front Genet 13. doi:10.3389/fgene.2022.958780.

Meng, T., X. Jing, Z. Yan, and W. Pedrycz. 2020. A survey on machine learning for data fusion. Information Fusion 57:115–129. doi:https://doi.org/10.1016/j.inffus.2019.12.001.

Mikolov, T., K. Chen, G. Corrado, and J. Dean. 2013. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

Myat Noe, S., T.T. Zin, P. Tin, and I. Kobayashi. 2023. Comparing State-of-the-Art Deep Learning Algorithms for the Automated Detection and Tracking of Black Cattle. Sensors 23. doi:10.3390/s23010532.

Okura, F., S. Ikuma, Y. Makihara, D. Muramatsu, K. Nakada, and Y. Yagi. 2019. RGB-D video-based individual identification of dairy cows using gait and texture analyses. Comput Electron Agric 165:104944. doi:https://doi.org/10.1016/j.compag.2019.104944.

Oliver, A., A. Odena, C.A. Raffel, E.D. Cubuk, and I. Goodfellow. 2018. Realistic Evaluation of Deep Semi-Supervised Learning Algorithms. Page in Advances in Neural Information Processing Systems. Curran Associates, Inc.

O'Mahony, N., S. Campbell, A. Carvalho, S. Harapanahalli, G.V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh. 2020. Deep Learning vs. Traditional Computer Vision. Pages 128–144 in Advances in Computer Vision. Springer International Publishing, Cham.

Overton, T.R., J.A.A. McArt, and D. V Nydam. 2017. A 100-Year Review: Metabolic health indicators and management of dairy cattle. J Dairy Sci 100:10398–10417. doi:https://doi.org/10.3168/jds.2017-13054.

Overton, T.R., and M.R. Waldron. 2004. Nutritional Management of Transition Dairy Cows: Strategies to Optimize Metabolic Health. J Dairy Sci 87:E105–E119. doi:https://doi.org/10.3168/jds.S0022-0302(04)70066-1.

Petersen, W.E. 1922. The Identification of the Bovine by Means of Nose-Prints1. J Dairy Sci 5:249–258. doi:https://doi.org/10.3168/jds.S0022-0302(22)94150-5.

Phyo, C.N., T.T. Zin, H. Hama, and I. Kobayashi. 2018. A Hybrid Rolling Skew Histogram-Neural Network Approach to Dairy Cow Identification System. Pages 1–5 in 2018 International Conference on Image and Vision Computing New Zealand (IVCNZ).

Qi, C.R., H. Su, K. Mo, and L.J. Guibas. 2016. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. arXiv e-prints arXiv:1612.00593. doi:10.48550/arXiv.1612.00593.

Qi, C.R., L. Yi, H. Su, and L.J. Guibas. 2017. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. Page in Advances in Neural Information Processing Systems. Curran Associates, Inc.

Qiao, Y., H. Kong, C. Clark, S. Lomax, D. Su, S. Eiffert, and S. Sukkarieh. 2021. Intelligent perception for cattle monitoring: A review for cattle identification, body condition score evaluation, and weight estimation. Comput Electron Agric 185:106143. doi:https://doi.org/10.1016/j.compag.2021.106143.

Qiao, Y., D. Su, H. Kong, S. Sukkarieh, S. Lomax, and C. Clark. 2019. Individual Cattle Identification Using a Deep Learning Based Framework. IFAC-PapersOnLine 52:318–323. doi:https://doi.org/10.1016/j.ifacol.2019.12.558.

Qiao, Y., D. Su, H. Kong, S. Sukkarieh, S. Lomax, and C. Clark. 2020. BiLSTM-based Individual Cattle Identification for Automated Precision Livestock Farming. Pages 967–972 in 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE).

Roche, J.R., N.C. Friggens, J.K. Kay, M.W. Fisher, K.J. Stafford, and D.P. Berry. 2009. Invited review: Body condition score and its association with dairy cow productivity, health, and welfare. J Dairy Sci 92:5769–5801. doi:https://doi.org/10.3168/jds.2009-2431.

Rodríguez Alvarez, J., M. Arroqui, P. Mangudo, J. Toloza, D. Jatip, J.M. Rodríguez, A. Teyseyre, C. Sanz, A. Zunino, C. Machado, and C. Mateos. 2018. Body condition estimation on cows from depth images using Convolutional Neural Networks. Comput Electron Agric 155:12–22. doi:https://doi.org/10.1016/j.compag.2018.09.039.

Rodríguez Alvarez, J., M. Arroqui, P. Mangudo, J. Toloza, D. Jatip, J.M. Rodriguez, A. Teyseyre, C. Sanz, A. Zunino, C. Machado, and C. Mateos. 2019. Estimating Body Condition Score in Dairy Cows From Depth Images Using Convolutional Neural Networks, Transfer Learning and Model Ensembling Techniques. Agronomy 9. doi:10.3390/agronomy9020090.

Salau, J., J.H. Haas, W. Junge, U. Bauer, J. Harms, and S. Bieletzki. 2014. Feasibility of automated body trait determination using the SR4K time-of-flight camera in cow barns. Springerplus 3:225. doi:10.1186/2193-1801-3-225.

Schokker, D., M. Poppe, J. ten Napel, I.N. Athanasiadis, C. Kamphuis, and R.F. Veerkamp. 2022. Rapid turnover of sensor data to genetic evaluation for dairy cows in the cloud. J Dairy Sci 105:9792–9798. doi:https://doi.org/10.3168/jds.2022-22113.

Schroff, F., D. Kalenichenko, and J. Philbin. 2015. FaceNet: A Unified Embedding for Face Recognition and Clustering. Page in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Sharma, A., L. Randewich, W. Andrew, S. Hannuna, N. Campbell, S. Mullan, A.W. Dowsey, M. Smith, M. Hansen, and T. Burghardt. 2024. Universal Bovine Identification via Depth Data and Deep Metric Learning. arXiv e-prints arXiv:2404.00172. doi:10.48550/arXiv.2404.00172.

Shen, W., H. Hu, B. Dai, X. Wei, J. Sun, L. Jiang, and Y. Sun. 2020. Individual identification of dairy cows based on convolutional neural networks. Multimed Tools Appl 79:14711–14724. doi:10.1007/s11042-019-7344-7.

Shi, W., B. Dai, W. Shen, Y. Sun, K. Zhao, and Y. Zhang. 2023. Automatic estimation of dairy cow body condition score based on attention-guided 3D point cloud feature extraction. Comput Electron Agric 206:107666. doi:https://doi.org/10.1016/j.compag.2023.107666.

Silva, F.F., G. Morota, and G.J. de M. Rosa. 2021. Editorial: High-Throughput Phenotyping in the Genomic Improvement of Livestock. Front Genet 12.

Smink, M., H. Liu, D. Döpfer, and Y.J. Lee. 2024. Computer Vision on the Edge: Individual Cattle Identification in Real-Time With ReadMyCow System. Pages 7056–7065 in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV).

Song, X., E.A.M. Bokkers, S. van Mourik, P.W.G. Groot Koerkamp, and P.P.J. van der Tol. 2019. Automated body condition scoring of dairy cows using 3-dimensional feature extraction from multiple body regions. J Dairy Sci 102:4294–4308. doi:https://doi.org/10.3168/jds.2018-15238.

Spoliansky, R., Y. Edan, Y. Parmet, and I. Halachmi. 2016. Development of automatic body condition scoring using a low-cost 3-dimensional Kinect camera. J Dairy Sci 99:7714–7725. doi:https://doi.org/10.3168/jds.2015-10607.

Steeneveld, W., P. Amuta, F.J.S. van Soest, R. Jorritsma, and H. Hogeveen. 2020. Estimating the combined costs of clinical and subclinical ketosis in dairy cows. PLoS One 15:e0230448-.

Sun, S., S. Yang, and L. Zhao. 2013a. Noncooperative bovine iris recognition via SIFT. Neurocomputing 120:310–317. doi:https://doi.org/10.1016/j.neucom.2012.08.068.

Sun, Y., X. Wang, and X. Tang. 2013b. Deep convolutional network cascade for facial point detection. Pages 3476–3483 in Proceedings of the IEEE conference on computer vision and pattern recognition.

Tharwat, A., T. Gaber, and A.E. Hassanien. 2014. Cattle Identification Based on Muzzle Images Using Gabor Features and SVM Classifier. Pages 236–247 in Advanced Machine Learning Technologies and Applications. Springer International Publishing, Cham.

Tsai, D.-M., and C.-Y. Huang. 2014. A motion and image analysis method for automatic detection of estrus and mating behavior in cattle. Comput Electron Agric 104:25–31. doi:https://doi.org/10.1016/j.compag.2014.03.003.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. ukasz Kaiser, and I. Polosukhin. 2017. Attention is All you Need. Page in Advances in Neural Information Processing Systems. Curran Associates, Inc.

Velez, J.F., A. Sanchez, J. Sanchez, and J.L. Esteban. 2013. Beef identification in industrial slaughterhouses using machine vision techniques. Spanish Journal of Agricultural Research 11:945–957.

Voulodimos, A.S., C.Z. Patrikakis, A.B. Sideridis, V.A. Ntafis, and E.M. Xylouri. 2010. A complete farm management system based on animal identification using RFID technology. Comput Electron Agric 70:380–388. doi:https://doi.org/10.1016/j.compag.2009.07.009.

Wang, H., J. Qin, Q. Hou, and S. Gong. 2020. Cattle Face Recognition Method Based on Parameter Transfer and Deep Learning. J Phys Conf Ser 1453:012054. doi:10.1088/1742-6596/1453/1/012054.

Wang, H., Y. Wang, Z. Zhou, X. Ji, Z. Li, D. Gong, J. Zhou, and W. Liu. 2018. CosFace: Large Margin Cosine Loss for Deep Face Recognition. CoRR abs/1801.09414.

Wang, R., R. Gao, Q. Li, C. Zhao, L. Ru, L. Ding, L. Yu, and W. Ma. 2024. An ultra-lightweight method for individual identification of cow-back pattern images in an open image set. Expert Syst Appl 249:123529. doi:https://doi.org/10.1016/j.eswa.2024.123529.

Weiss, K., T.M. Khoshgoftaar, and D.D. Wang. 2016. A survey of transfer learning. J Big Data 3. doi:10.1186/s40537-016-0043-6.

Wu, Y., H. Guo, Z. Li, Q. Ma, Y. Zhao, and A. Pezzuolo. 2021. Body Condition Score for Dairy Cows Method Based on Vision Transformer. Pages 37–41 in 2021 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor).

Xiao, J., G. Liu, K. Wang, and Y. Si. 2022. Cow identification in free-stall barns based on an improved Mask R-CNN and an SVM. Comput Electron Agric 194:106738. doi:https://doi.org/10.1016/j.compag.2022.106738.

Xu, B., W. Wang, L. Guo, G. Chen, Y. Li, Z. Cao, and S. Wu. 2022. CattleFaceNet: A cattle face identification approach based on RetinaFace and ArcFace loss. Comput Electron Agric 193:106675. doi:https://doi.org/10.1016/j.compag.2021.106675.

Yang, Z., H. Xiong, X. Chen, H. Liu, Y. Kuang, and Y. Gao. 2019. Dairy Cow Tiny Face Recognition Based on Convolutional Neural Networks. Pages 216–222 in Biometric Recognition. Springer International Publishing, Cham.

Yao, L., Z. Hu, C. Liu, H. Liu, Y. Kuang, and Y. Gao. 2019. Cow face detection and recognition based on automatic feature extraction algorithm. Page in Proceedings of the ACM Turing Celebration Conference - China. Association for Computing Machinery, New York, NY, USA.

Yukun, S., H. Pengju, W. Yujie, C. Ziqi, L. Yang, D. Baisheng, L. Runze, and Z. Yonggen. 2019. Automatic monitoring system for individual dairy cows based on a deep learning framework that provides identification via body parts and estimation of body condition score. J Dairy Sci 102:10140–10151. doi:https://doi.org/10.3168/jds.2018-16164.

Zhao, K., M. Zhang, W. Shen, X. Liu, J. Ji, B. Dai, and R. Zhang. 2023. Automatic body condition scoring for dairy cows based on efficient net and convex hull features of point clouds. Comput Electron Agric 205:107588. doi:https://doi.org/10.1016/j.compag.2022.107588.

Zin, T.T., S. Misawa, M.Z. Pwint, S. Thant, P.T. Seint, K. Sumi, and K. Yoshida. 2020a. Cow Identification System using Ear Tag Recognition. Pages 65–66 in 2020 IEEE 2nd Global Conference on Life Sciences and Technologies (LifeTech).

Zin, T.T., C.N. Phyo, P. Tin, H. Hama, and I. Kobayashi. 2018. Image technology based cow identification system using deep learning. Pages 236–247 in Proceedings of the international multiconference of engineers and computer scientists.

Zin, T.T., P.T. Seint, P. Tin, Y. Horii, and I. Kobayashi. 2020b. Body Condition Score Estimation Based on Regression Analysis Using a 3D Camera. Sensors 20. doi:10.3390/s20133705.

Zou, J., M. Huss, A. Abid, P. Mohammadi, A. Torkamani, and A. Telenti. 2019. A primer on deep learning in genomics. Nat Genet 51:12–18. doi:10.1038/s41588-018-0295-5.

# TABLES AND FIGURES

**Table 1.1.** Summary of the main concerns and feature extraction methods proposed for BCS prediction using computer vision.

| Works | Image type | Validation concerns | Automation concerns | Features extracted |
|---|---|---|---|---|
| (Bewley et al., 2008; Battiato et al., 2010; Azzaro et al., 2011) | 2D | No independent test set | Manual keypoint detection | Shape descriptors from anatomical keypoints |
| (Halachmi et al., 2008, 2013) | 2D | No independent test set | | Residuals of fitted parabola |
| (Bercovich et al., 2013) | 2D | | Manual image selection | Shape descriptors from anatomical keypoints, and FD[1] of tailhead contour |
| (Huang et al., 2019) | 2D | Possible temporal bias between training and testing data | | CNN[2] |
| (Li et al., 2019) | 2D | Possible temporal bias between training and testing data | Manual image selection | CNN[2] |
| (Wu et al., 2021) | 2D | Possible temporal bias between training and testing data | | CNN[2] and Vision Transformers |
| (Salau et al., 2014) | 3D | Only correlation analysis | Manual image selection | Shape descriptors from anatomical keypoints |
| (Kuzuhara et al., 2015) | 3D | No independent test set | Manual keypoint detection and image collection | Geodesic distances between anatomical keypoints |
| (Fischer et al., 2015) | 3D | | Manual keypoint detection | Principal components of 3D point coordinates |
| (Song et al., 2019) | 3D | | Manual image collection and empirical processing parameters | Shape descriptors from anatomical keypoints |
| (Liu et al., 2020) | 3D | | Empirical image processing parameters | Shape descriptors from anatomical keypoints |
| (Spoliansky et al., 2016) | 3D | | Empirical image processing parameters | Shape descriptors extracted from pixel region empirically found |
| (Hansen et al., 2015, 2018) | 3D | No independent test set | | Angularity of 3D surface |
| (Zin et al., 2020b) | 3D | No extreme examples | | Roughness parameters from 3D surface |
| (Rodríguez Alvarez et al., 2018, 2019) | 3D | Few extreme examples | | CNN[2] |
| (Yukun et al., 2019) | 3D | | Manual image collection | CNN[2] |
| (Zhao et al., 2023) | 3D | | | CNN[2] |
| (Shi et al., 2023) | 3D | | | Attention-guided point cloud feature extraction |

[1]FD = fourier descriptors
[2]CNN = convolutional neural network

# CHAPTER TWO: USING DORSAL SURFACE FOR INDIVIDUAL IDENTIFICATION OF DAIRY CALVES THROUGH 3D DEEP LEARNING ALGORITHMS

## ABSTRACT

Advances in machine learning techniques have allowed the development of computer vision systems (**CVS**) that can accurately predict several phenotypes of interest for livestock operations. In this context, 3-dimensional (**3D**) images taken from a top-down view are particularly useful for estimating body condition score, growth development, and body biometrics in cattle. Frequently, such CVS rely on identification (**ID**) systems, such as electronic tags, as a way to match animal ID and the predicted phenotype. However, the same 3D images used to predict body weight and other animal biometrics could be adopted for animal recognition as well. Such alternative would optimize CVS to recognize animal ID and monitor growth development simultaneously while leveraging the same hardware infrastructure. Furthermore, this strategy could be used to recognize animals with similar color patterns. Nonetheless, growing animals are continuously changing body shape, which could limit its use as an invariant feature for pattern recognition. Thus, the objectives of this study were: (1) to compare algorithms for different 3D object representations to identify individual animals; and (2) to evaluate how short-term changes in body shape due to animal growth affect the predictive performance of these algorithms. For objective 1, the algorithms were trained (n = 4,558) and tested (n = 1,139) using images from 38 Holstein calves. For objective 2, we designed three different experiments using images (n = 2,347) from five Holstein calves taken over six weeks during their growing period, always training and testing on different weeks. Each experiment evaluated how changing a different parameter of the image capturing procedure affected the predictive ability of the trained algorithms. In the first

experiment, we varied the total number of images per animal in the training set; in the second experiment, we varied the number of weeks while keeping a fixed number of images in the training set; and in the third experiment, we skipped weeks between images in the training and test sets. The $F_1$ score for objective (1) was up to 0.804 when testing with the last frames of each video, and up to 0.959 when using random frames for testing. For objective (2), the $F_1$ score was up to 0.947 for the first experiment when using 130 images per animal; up to 0.979 for the second experiment when using all five weeks; and up to 0.917 when not skipping weeks between training and testing. These results show that deep learning algorithms can be used to identify individual animals through their dorsal area 3D surfaces, and, from our experiments using calves in their growing period, that they are robust enough to account for changes in body shape and size, making them a promising tool for animal recognition during growth.

## INTRODUCTION

Deep learning techniques have gained great popularity in the field of computer vision in recent years due to their impressive performance in tasks such as image classification, object detection, and semantic segmentation (Voulodimos et al., 2018). Deep learning allows machine learning models to learn abstract feature representations of the input data and perform automatic feature extraction when exposed to large amounts of data (LeCun et al., 2015). Such advances in deep learning and computer vision, and particularly in the use of depth sensing cameras, have enabled the development of systems that capture animal phenotypes such as body condition, body weight, lameness, behavior traits, and more (Fernandes et al., 2020). In order to capture and use animal-level phenotypes, implementing a system to identify individual animals is vital. These systems can be manual, such as ear tags, or automated, such as radio-frequency identification

(**RFID**) (Voulodimos et al., 2010). However, implementing manual identification or RFID systems in large scale operations can be labor-intensive, prone to human error and fraud, costly, and invasive for the animals, as it requires manually placing RFID tags on each animal.

In this context, using computer vision techniques to implement both animal identification and phenotyping into one single integrated system can be beneficial, as it could limit the use of external accessories attached to animals, leverage the same hardware infrastructure, and therefore address most of the issues related to RFID systems. Moreover, computer vision systems (**CVS**) could be a robust alternative to track animals along the food supply chain, allowing the development of traceability programs with high degree of security as found in blockchain systems (Casino et al., 2019). Recent studies have proposed the use of Red, Green, Blue (**RGB**) images to identify animals based on their unique coat color patterns in different species by using 2-dimensional (**2D**) convolutional neural networks (**CNN**). Andrew et al. (2017) and Bello et al. (2020) used 2D CNNs to identify Holstein cows using top-view images of their back, Yao et al. (2019) used detection and classification 2D CNNs to detect and identify Holstein cows using images of their faces, Yukun et al. (2019) used RGB and depth images to automatically identify Holstein cows and estimate their body condition scores, and Hansen et al. (2018) proposed their own 2D CNN to individually identify pigs using images of their faces. However, these approaches require that individual animals have different coat color patterns, so they would likely fail to differentiate animals with similar colors patterns, or certain animal breeds that have little color distinction between individuals.

As an alternative to RGB images, different 3-dimensional (**3D**) data representations can be used to classify objects. For example, depth images, despite being virtual representations of 3D surfaces, can be used along with 2D CNNs to perform classification tasks, because they are

actually 2D images where each pixel contains a value representing the distance between the physical point at that pixel and the camera sensor. Additionally, 3D CNNs and other neural network architectures have been recently proposed to work with other 3D representations, such as voxels (Maturana and Scherer, 2015), octrees (Wang et al., 2017), and point clouds (Qi et al., 2016). These representations can prove beneficial in classifying objects whose 3D shape is more relevant than their color, as showed by Aijazi et al. (2013) when segmenting urban scenes, and Soilán Rodríguez et al. (2019) when classifying data acquired with Airborne Laser Scanning systems, for example. Such tasks, however, can be challenging when working with objects that quickly change their shapes over time, such as animals during their growing stage of life.

The current study aims to evaluate the predictive ability of deep neural networks to identify individual calves based on the shape of their dorsal region, using different 3D representations as input data. Additionally, we evaluated the robustness of the tested algorithms to perform this task as body shape changes due to animal growth. To accomplish that, we (1) compared algorithms for different 3D object representations to identify individual animals by using images collected in the same period of time; and (2) evaluated how short-term changes in body shape due to animal growth affect the predictive performance of these algorithms.

## MATERIAL AND METHODS

This study was split into two objectives, as previously mentioned. For the first one, we compared the performance of five neural network architectures on identifying individual calves by using different 3D data representations. Three of them were 2D CNNs using depth images as inputs, and the other two were a 3D CNN using voxels as inputs, and a combination of multi-layer perceptrons using point clouds as inputs. For the second objective, the same five neural network architectures were assessed on identifying individual calves in different periods of time during

their growing stage, in order to evaluate how changes in body shape would affect the predictive performance of these algorithms.

Datasets

For the first objective, videos from 38 pre-weaned Holstein dairy calves with ages varying from two to eight weeks, and body weight (**BW**) of 57.0 ± 14.7 kg (average ± standard deviation (**SD**)), housed at the Emmons Blaine Dairy Cattle Research Center (Arlington, WI), were recorded during a single week. A Kinect V2 sensor (Microsoft; Redmond, WA) was used, which has an RGB camera (resolution of 1920 × 1080 pixels), a depth sensor (resolution of 512 × 424 pixels), and a microphone array. The 38 videos were recorded from a top-down view, and each contained a single calf, as they were recorded separately while weighing each animal individually. All videos were recorded using Kinect for Windows SDK 2.0 (Microsoft; Redmond, WA) installed on a laptop locally operated by a person who manually started recording as soon as the calf was positioned on the scale, and stopped recording when the weighing process was concluded for that calf. The length of the videos varied from 15 to 69 seconds, from which frames from the depth stream were extracted at a rate of four frames per second (**FPS**). This resulted in a total of 5,764 depth frames with a resolution of 512 × 424 pixels, each pixel representing the distance from the object to the camera sensor in millimeters.

For the second objective, 30 videos from five calves with ages varying from four to eight weeks, and BW of 63.8 ± 6.7 kg (average ± SD), housed at the Dairy Cattle Research Center (**DCRC**; Madison, WI), were recorded using the same Kinect V2 sensor (Microsoft; Redmond, WA) from a top-down view, and the same recording procedures as in the first objective. Each calf had the videos recorded separately once a week for six weeks, with video recording lengths between 18 and 80 seconds. Depth frames were then extracted at a rate of two FPS, resulting in a

total of 2,347 frames with a resolution of 512 × 424 pixels, each pixel representing the distance from the object to the camera sensor in millimeters.

Data Preprocessing

Data preprocessing was performed for each acquired frame in each dataset, and it involved four steps, in the following order: (1) background removal, (2) point cloud generation, (3) point cloud augmentation, and (4) occupancy grid generation. The four steps are described in the following subsections (*Background Removal*, *Point Cloud Generation*, *Point Cloud Augmentation*, *Occupancy Grid Generation*).

Background Removal

In order to remove background pixels from the captured depth images, a network based on the Mask R-CNN framework (He et al., 2017) was implemented to automatically detect and retain all pixels containing a calf. We only considered as part of the calf the region between the tail and the neck of the animal. The Mask R-CNN network was trained using 584 depth images manually segmented according to this standard, as shown in Figure 2.2(b), where pixels containing the calf appear in white. Some of the frames captured from the original videos did not contain a calf, resulting in 5,697 frames for the first objective, and 2,295 for the second. The trained network for calf segmentation achieved an intersection over union of 0.932 on an independent test set.

Point Cloud Generation

The pixels detected as containing a calf were converted to a set of points in a 3-dimensional coordinate system (a point cloud). For each pixel ($i$, $j$) containing a depth value $d$, a point ($x_p$, $y_p$, $z_p$) was created with values ($x_p$, $y_p$, $z_p$) = ($j$, $i$, $d$). This resulted in a point cloud with the number of points equal to the number of pixels that were part of a calf in the original frame. Outlier points were then removed based on their Z-axis coordinates, or depth value, in order to prevent the

inclusion of background pixels due to segmentation errors. A value was considered an outlier if it was more than three scaled median absolute deviations (**MAD**) away from the median. For a random vector *X* with *N* scalar observations, the MAD is defined as follows:

$$MAD = median(|Xi - median(X)) \quad \text{(Eq. 2.1)}$$

for $i = 1, 2, \ldots, N$

The scaled MAD is defined as $k \times$ MAD, where $k \approx 1.4826$ is a constant scale factor that depends on the distribution (Rousseeuw and Croux, 1993). In this case, we operated under the assumption that the Z-axis values were normally distributed.

Point Cloud Augmentation

The generated point cloud was then augmented by randomly rotating, scaling, and applying jitter to the point coordinates. Image augmentation is a technique used to avoid overfitting and add robustness to 2D convolutional networks (Perez and Wang, 2017). Point cloud augmentation, however, is a similar technique with some important differences. The main difference is in the rotation process: point cloud augmentation allows the objects to be rotated around any of the three axes, as opposed to image augmentation, where the image can only be rotated around a single axis (the one pointing towards the image plane). In this study, the point clouds were rotated around their Z-axis by a random angle between 0 and 360 degrees, the coordinates were scaled by a random factor between 0.98 and 1.02, and a 1% jitter was applied to each point. These values were chosen arbitrarily. Applying these transformations introduced noise to the data, avoiding overfitting and making the trained models more robust to rotation.

Occupancy Grid Generation

The point cloud resulting from the augmentation step was then converted into an occupancy grid by splitting the coordinate space of the points into 32 cells on each axis. For each point ($x_p$,

$y_p$, $z_p$), the coordinate values of the containing cell in the grid space ($x_{cell}$, $y_{cell}$, $z_{cell}$) were calculated as follows:

$$x_{cell} = \min\left(\left\lfloor \frac{x_p - min_x}{max_x - min_x} \times 32 \right\rfloor, 31\right)$$

$$y_{cell} = \min\left(\left\lfloor \frac{y_p - min_y}{max_y - min_y} \times 32 \right\rfloor, 31\right) \quad \text{(Eq. 2.2)}$$

$$z_{cell} = \min\left(\left\lfloor \frac{z_p - min_z}{max_z - min_z} \times 32 \right\rfloor, 31\right)$$

Values $min_x$ and $max_x$ were the minimum and maximum $x_p$ values in the point cloud, and likewise for *y* and *z*, resulting in values in the range [0, 31] for each cell coordinate. Based on the cell coordinates of each point, the 32 × 32 × 32 grid was then filled with ones or zeros depending on whether the corresponding cell contained at least one point of the original point cloud (Figure 2.1). Occupancy grids can serve as a more regular 3D representation of the data in comparison to point clouds, with grid cells contained in a discrete domain as opposed to the continuous nature of point coordinates in point clouds. Such regularization can help machine learning systems learn more efficiently than with more irregular formats such as raw point clouds, by adopting 3D convolutional neural networks, for example (Maturana and Scherer, 2015). Figure 2.2 shows an example of the step-by-step process of transforming a depth frame into an occupancy grid.

Training and Test Sets

For the first objective, two different approaches were used to split the dataset into training and test sets. In the first approach, 5,697 frames were randomly split into training (n = 4,558) and test (n = 1,139) sets, corresponding to 80% and 20% of the total dataset, respectively, without necessarily maintaining class proportions between training and test sets. This process was repeated 10 times, generating 10 different random dataset splits that were used to calculate an average final performance metric. The randomization was done at the level of the entire 38 videos, generating

slightly different class proportions for each permutation. In the second approach, the frames from each video were split chronologically based on their positions in the video, separating the first 80% frames for training and last 20% for testing. We used the second approach to minimize similarities between the training and test sets, as adjacent frames tend to be similar to each other (see Figure 2.3).

For the second objective, three different experiments were designed, and the dataset was split accordingly. The first experiment consisted of evaluating how the number of frames used in training would affect the predictive performance of the algorithms. For that, random samples of 20, 40, 70, 100, 130, and 154 images per animal were used for training, all from the first and second weeks, and a fixed set of 319 images from the third week was used for testing.

In the second experiment, we evaluated how increasing the number of consecutive weeks used for training affected the performance of the algorithms on the immediate following week, while keeping the same total amount of images per animal. We used 80 images per animal for training (resulting in a total of 400 images), and tested on images from the following week, such that the size of the test set varied according to the week, but the training set size remained constant. A total of ten dataset splits were created for this experiment, grouping them according to the total number of weeks used for training, and calculating an average performance for each group (Table 2.1).

Finally, in the third experiment, we evaluated the effect of increasing the time interval between the training and test sets on the prediction quality of the tested algorithms. In this context, we defined four time intervals in relation to weeks after training: zero (testing on images from the subsequent week), one, two, and three weeks. For training, we used two consecutive weeks and 80 images per animal (resulting in a total of 400 images) for each split. Ten splits were created for

this experiment, grouped according to the interval between the training and test sets. The size of the test set varied according to the week used for testing (Table 2.2). Table 2.3 provides an overview of the three experiments performed for the second objective.

Data Representation and Algorithms

The algorithms were chosen based on the data representation used as input. Algorithms able to analyze 2D depth images (Simonyan and Zisserman, 2014; Szegedy et al., 2016; Chollet, 2017), point clouds (Qi et al., 2016), and occupancy grids (Maturana and Scherer, 2015) were selected. All algorithms were implemented in Python, using TensorFlow (Abadi et al., 2016) for implementing PointNet, TensorFlow and Keras (Chollet, 2015) for implementing VGG16, Inception v3 and Xception, and Theano (The Theano Development Team et al., 2016) and Lasagne (Dieleman et al., 2015) for implementing VoxNet.

Depth Images – VGG16, Inception v3, and Xception

To generate depth images from the extracted video frames, the data was processed using only the first preprocessing stage (*Background Removal*). The resulting mask was applied to the pixel-based depth values, setting every pixel not contained in the mask to zero. Outliers were then identified using the method presented in the *Point Cloud Generation* subsection, and their corresponding values were set to zero. The final depth image consisted of a matrix of size 424 × 512 containing the depth values of relevant pixels, or zero for pixels considered part of the background.

These depth images were then used as the input to three different deep neural network (**DNN**) architectures: VGG16 (Simonyan and Zisserman, 2014), Inception v3 (Szegedy et al., 2016), and Xception (Chollet, 2017). For all three DNNs, the last fully-connected (**FC**) layer of the original architecture was removed, and all the other layers were initialized with weights from

the respective networks trained using ImageNet (Deng et al., 2009), an open image dataset containing more than 1 million examples of diverse objects and environments, ranging from wild and farm animals to vehicles, airplanes, and housewares, for example. Such strategy was defined as Transfer Learning (Weiss et al., 2016), and it accelerates the training process as the network weights are initialized with values optimized for a large generic image dataset such as ImageNet, instead of being initialized with random values. This technique helped our new networks learn generic features, such as textures, edges, corners, and shapes, previously learned in a different task domain using a much larger dataset.

The VGG16-based network was extended with a FC layer of size 2,048 and a Rectified Linear Unit (**ReLU**) activation function (Nair and Hinton, 2010), followed by a final FC layer of size *n* and softmax activation function, where *n* is the number of classes for each objective (n = 38 for the first objective and n = 5 for the second objective).

The Inception v3- and Xception-based DNNs were extended with a global average pooling layer as described by Lin et al. (2013), followed by a FC layer of size 1,024 and ReLU activation function, and a final FC layer of size *n* and softmax activation function, similarly to the VGG16-based approach.

For each DNN, the training process was split into two consecutive stages: feature extraction and fine-tuning. In the feature extraction stage, the DNN was trained for 200 epochs keeping the weights of all but the last two FC layers frozen. This allowed features previously learned through Transfer Learning to be used and retained. In the fine-tuning stage, weights from earlier layers were unfrozen, and the network was trained for 400 epochs with a smaller learning rate, allowing it to further learn features that are more specific to our context.

The VGG16-based network was trained using RMSProp (Hinton et al., 2012) with a learning rate of $2 \times 10^{-5}$ in the feature extraction stage and $1 \times 10^{-5}$ in the fine-tuning stage. The Inception v3-based network was trained using RMSProp with a learning rate of $1 \times 10^{-3}$ in the feature extraction stage, and Stochastic Gradient Descent (Robbins and Monro, 1951) with a learning rate of $1 \times 10^{-4}$ and momentum of 0.9 (Qian, 1999) in the fine-tuning stage. The Xception-based network was trained using Adam (Kingma and Ba, 2014) with a learning rate of $1 \times 10^{-3}$ in the feature extraction stage and $1 \times 10^{-5}$ in the fine-tuning stage.

Point Cloud – PointNet

From the point clouds generated by applying the first three preprocessing stages described in the *Data Preprocessing* section, the k-means clustering algorithm was used to separate the 3D points into 2,048 clusters. The centroids of these clusters were then grouped into a new point cloud and used as the input to a network based on the full PointNet architecture (Qi et al., 2016). We decided to use point clouds of size 2,048 because PointNet was designed, trained, and validated using the ModelNet40 dataset (Wu et al., 2015), which contains point clouds of size 2,048. The last FC layer of the original PointNet architecture was modified to have *n* nodes, where *n* is the number of classes for each objective, as before. The network was trained for 250 epochs using Adam (Kingma and Ba, 2014) with an initial learning rate of $1 \times 10^{-3}$, a momentum of 0.9, and exponential learning rate decay of 0.7 every 200,000 steps.

Occupancy Grid (Voxel) – VoxNet

The occupancy grids generated from applying all four preprocessing stages described in the *Data Preprocessing* section, also known as voxels, were used as the input to a network based on the VoxNet architecture (Maturana and Scherer, 2015). The grid size was defined as $32 \times 32 \times 32$, the same as proposed in the original VoxNet article (Maturana and Scherer, 2015). The last FC

layer of the architecture was modified to have *n* nodes, the number of classes for each objective. The network was trained for 400 epochs using Stochastic Gradient Descent with a learning rate of $1 \times 10^{-3}$, a momentum of 0.9, and L2 norm regularization of 0.001 applied to the loss function.

Evaluation Metrics

To evaluate and compare the prediction quality of all algorithms, the accuracy, precision, recall, and $F_1$ score were calculated for each class as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad \text{(Eq. 2.3)}$$

$$Precision = \frac{TP}{TP + FP} \quad \text{(Eq. 2.4)}$$

$$Recall = \frac{TP}{TP + FN} \quad \text{(Eq. 2.5)}$$

$$F_1 score = \frac{2 \times TP}{2 \times TP + FP + FN} \quad \text{(Eq. 2.6)}$$

where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

The mean values across all classes were then calculated and used to compare the algorithms (macro averaging), and the final $F_1$ score was calculated as the mean of the class-wise $F_1$ scores. Precision, Recall, and $F_1$ score are important metrics to evaluate classification tasks. They can be more informative than the accuracy in a context of imbalanced data, where the number of images corresponding to each class varies significantly.

## RESULTS AND DISCUSSION

Comparing Algorithms and 3D Representations

The first objective consisted of comparing different algorithms and 3D representations to identify individual calves using their dorsal surfaces. The results discussed in this subsection are

related to the approach of using a chronologically ordered split of the frames, in order to prevent overoptimistic results from using adjacent frames in the training and test sets (refer to section *Training and Test Sets* for details). Preventing biased evaluation results is an important step in any Artificial Intelligence system, as the main goal of the evaluation process is to try to anticipate how the algorithm will perform when facing real-world scenarios. Thus, when working with algorithms designed to generate predictions on images that will be captured in the future, it is critical to use the earliest captured images as the training set, and include only the latest captured images in the test set, in order to achieve more realistic results. A report of all calculated $F_1$ scores can be found in Table 2.4. Using random images in a sequence for training and testing generates higher, overestimated $F_1$ scores, when compared to a more realistic scenario of the test set containing only the last frames of the original videos. For example, when using Xception, the $F_1$ score decreases from 0.959 to 0.804 when using the chronological order approach, which is a more realistic approximation of how that network would perform on future images.

The 2D CNN approaches achieved $F_1$ scores of 0.718, 0.750, and 0.804 with the VGG16-, Inception v3- and Xception-based networks, respectively. These results were consistent with the results reported in the original Xception publication (Chollet, 2017), with Xception performing better than VGG16 and Inception v3 on the ImageNet and JFT datasets. This improvement comes from making use of inception modules (Szegedy et al., 2015) and introducing depthwise separable convolutions (Chollet, 2017).

The point cloud approach using a PointNet-based network achieved an $F_1$ score of 0.429, which is the lowest of all the approaches for this objective. This is probably because, before being fed to the network, the original point clouds resulted from the preprocessing step were reduced from approximately 30,000 to 2,048 points. This downsampling was much stronger than the one

performed in the original PointNet article (Qi et al., 2016), which proposed a downsampling of the point clouds in the ModelNet40 dataset from 2,048 to 1,024 points. This may have caused our network to miss important nuances from the surface of the calves, which are necessary to uniquely identify them. This evidence was supported when we tested this PointNet approach using only 1,024 points, and it resulted in a further $F_1$ score drop to 0.318. The PointNet architecture was designed to recognize objects that are structurally very different from each other, such as cars, tables, and airplanes. When distinguishing such different objects, it is not significantly detrimental to make use of fewer points, because the network learns how to use a collection of critical points to summarize the shapes (Qi et al., 2016), and the summarized shapes are usually very different from each other. However, this architecture may not be suitable for objects that are very similar in shape, and which the difference between classes is in small details and nuances, such as in the case of identification of calves.

Using voxels as input, the VoxNet-based network achieved an $F_1$ score of 0.656, which is superior to the results achieved using PointNet, but still below any of the $F_1$ scores achieved using 2D CNNs. VoxNet performed better than PointNet mostly because the voxels used in this study had a higher dimensionality than the 2,048-sized point clouds. They were contained in grids of 32 × 32 × 32 cells, so a total of 32,768 cells each. However, 2D CNNs performed better than VoxNet, possibly because they contain more trainable parameters, allowing them to represent more complex functions and to learn and extract greater levels of details from the inputs. Extracting high-dimensional feature representations appears to be beneficial for individual calf recognition, as shown in the results. Additionally, the 2D CNNs used were pretrained using the ImageNet dataset (Deng et al., 2009), which helped them learn more generic features before being trained with our datasets, further improving their results in comparison to PointNet and VoxNet, which

did not undergo any pretraining step. It is worth noting that other publicly available datasets could be used for pretraining the 2D CNNs, such as datasets containing exclusively images of animals, for example, which would be more similar to the input images used in this study. However, we could not find publicly available weights for Inception v3, VGG16, or Xception architectures pre-trained using such animal datasets, and training those networks from scratch requires significant amount of time and computational resources, especially when using large image datasets (Simonyan and Zisserman, 2014; Szegedy et al., 2016; Chollet, 2017). Future research could be done to evaluate how the choice of pretraining dataset for transfer learning affects the predictive performance of neural networks for animal identification, assessing the trade-off between using a dataset that is more similar to the one used in the final task, as opposed to a larger, more general dataset such as ImageNet.

Networks that contain more trainable parameters, combined with higher-dimensional inputs, perform better in the task of calf identification using 3D images of their dorsal surfaces, as they can capture more subtle variations in their shapes. Such nuances can be helpful when trying to uniquely identify individuals. The depth images used in this study contained approximately 30,000 foreground pixels, the voxel grids contained 32,768 cells, and the point clouds used for PointNet contained only 2,048 points. The Xception-based network used had 23 million parameters, while the PointNet-based network had just 3.5 million, and the VoxNet-based network had less than 1 million. This possibly explains why the Xception-based network was the best performing algorithm in this task when compared to point cloud- and voxel-based representations and architectures (PointNet and VoxNet), and these results agree with another work in the literature that performs similar comparisons for human face recognition (Pini et al., 2021).

Although 2D CNNs performed better in this specific setting where all videos were taken from a top-down view of the animals, 2D depth images can only hold surface information about one specific view of an object. Conversely, 3D representations such as point cloud and voxel bring the possibility to merge multiple views of the same object into one single instance (Narayanan et al., 1998; Seitz et al., 2006), and hold volumetric information about an object. This enables deep learning algorithms to perform classification and identification tasks using multi-view 3D representations, which contain a more robust and accurate depiction of the real object, possibly leading to better results (Gezawa et al., 2020). Although in this study we only used cameras positioned in a single fixed angle, it would be possible to take pictures from multiple different angles and build a full 3D volumetric representation of the calves. Moreover, while 2D representations are limited to rotations around a single axis, 3D representations can be augmented by rotating the object around all three axes, or even by implementing an automated data augmentation policy, generating more realistic unseen versions of the same animal (Cheng et al., 2020).

The networks employed in this study were trained using images of the animals taken exclusively from a top-down view, and thus they can only effectively identify individual animals in new images taken from that same angle. Alternatively, if the experiment included images taken from different angles, it would be necessary to utilize a separate 2D image augmentation process for each group of depth images taken from the same angle. For example, if four synchronized cameras were positioned to take pictures of the same animal from different angles, they would generate four 2D depth images per time point and animal, each undergoing a separate augmentation process. However, when using 3D representations such as voxels and point clouds, one single instance could represent the whole 3D animal by assembling images taken from

different angles and reconstructing a full 3D model of the animal, as described by Narayanan et al. (1998), allowing for more effective augmentation approaches, such as the ones reported by Hahner et al. (2020) and Cheng et al. (2020). In this case, four pictures taken from synchronized cameras would result in a single 3D voxel or point cloud. Such process could enhance the performance of the trained networks and yield superior results, as they could better generalize to a wider variety of camera angles and animal positions in this setting where images are captured from different views simultaneously (Cheng et al., 2020; Gezawa et al., 2020; Hahner et al., 2020). In this situation, 3D representations and networks could prove more useful than their 2D counterparts, despite achieving worse results in the context of our study.

### Evaluating How Short-Term Changes in Body Shape Affects the Predictive Performance of the Algorithms

Several situations can cause fast body shape changes in a short period of time, such as growth development in young animals (Cominotte et al., 2020), or body tissue mobilization to supply energy demands in early lactating dairy cows (Dórea et al., 2017). Monitoring an animal throughout a long period of its life, including such periods of body shape change, can have serious implications in animal disease control and food traceability, by making it possible to backtrack disease outbreaks in a farm, and ensure that products derived from that animal follow local sanitary regulations (Awad, 2016). However, such changes could hinder the predictive performance of the evaluated algorithms, as an individual in an image captured in the future could look different from when previous images were captured and used for training the animal identification algorithms. Nevertheless, as is the case for human faces (Park et al., 2010), there might be unique biometric features and landmarks on the body shape of the animals that remain proportional and recognizable regardless of the overall change in body size and shape. If the utilized algorithms are not robust

enough to identify these features and account for body variations, they would have to be retrained frequently during these periods of intense body shape change. Frequently retraining such convolutional neural networks could be extremely costly and labor intensive, as they would require a large dataset of new labeled images (LeCun et al., 2015), and the labeling process would consist of manually assigning each image to the correct animal. Because of that, it is important to evaluate if the utilized algorithms can still accurately identify individual animals even as they experience body changes. Thus, the second objective of this study was to evaluate how short-term changes in body shape affected the predictive performance of the assessed algorithms.

For the first experiment, which consisted of evaluating how the number of training images affected the predictive performance of the algorithms, the best results were achieved using the VoxNet-based network. Since we only used the first three weeks of data for this experiment, the simpler VoxNet architecture was sufficient, and the greater number of parameters and complexity of the Xception architecture did not translate into better results in this case. As shown in Table 2.5, increasing the number of training images per animal from 20 to 100 improved the $F_1$ score from 0.734 to 0.944. This shows that deep neural networks usually benefit from having more images available during training, so they can learn more intricate patterns and diverse examples from the training set, which help them better generalize to new data (LeCun et al., 2015). In our experiment, using more than 100 images per animal did not further improve the algorithms' performance significantly, probably due to the uniformity of our dataset, with all images captured from the same view and location. Therefore, including more images possibly just added more redundancy to the training set.

For the second experiment, which consisted of evaluating how the number of consecutive weeks used for training influenced the performance of the algorithms on the subsequent week, the

best results were achieved using the Xception-based network. Table 2.6 shows that, even as the training set size remained constant, including more weeks slightly increased the $F_1$ score of this network, and the highest score improvement happened when adding a fourth week to the training set. However, the PointNet- and VoxNet-based networks did not benefit from adding more weeks. This is likely because the Xception-based network, with a great number of parameters and trained using high resolution depth images, was the only network complex enough to capture useful information contained in more than two weeks concurrently. For the VoxNet-based network, with fewer parameters, and the PointNet-based network, using relatively low-density point clouds, additional weeks possibly just translated into more noise added to the training set, rather than contributing to better results.

Nevertheless, these results show that even by using just two weeks, both VoxNet and Xception could learn sufficient patterns from the 3D shape of the calves to identify them on the next week. This means that it might not be necessary to accumulate a long history of labeled images before being able to identify animals in new images, even if those animals are in a growing stage. According to the outcomes of our experiments, depth images of the back of calves as young as three weeks old can be used to train networks able to identify them during the subsequent week, showing that 3D deep learning systems can be used to monitor animals from a very early stage of life. Monitoring animals from an early stage is key for disease control as there is a high incidence of infectious diseases during that period (Marcé et al., 2010; Cho and Yoon, 2014). Thus, such identification and monitoring systems can help farmers make better management decisions to minimize the occurrence of such diseases and prevent the high economic losses associated with them (Kaneene and Scott Hurd, 1990; Esslemont and Kossaibati, 1999).

For the third experiment, which consisted of evaluating how skipping weeks between training and test sets affected the predictive performance of the algorithms, the best results were achieved, again, using the Xception-based network. Table 2.7 shows how skipping weeks between training and testing affected the algorithms' predictive performance. Using the Xception-based network, the $F_1$ score decreased from 0.917 to 0.846 when skipping one week. However, skipping more weeks did not further decrease the $F_1$ score of this network considerably, showing that it might be possible to skip up to three weeks between training the network and identifying calves in new images without significantly affecting its predictive performance. This is evidence that the network might be learning unique biometric features on the body surfaces that remain proportional as the animals grow. Thus, although labeling new images and retraining the network every week would yield the best results, it is still viable to train the network once and use it to identify calves on images taken three weeks later without a significant effect on the predictive ability.

By retraining the network only every three weeks, it is possible to reduce the time and effort dedicated to labeling new images and performing the network training routine. Building upon the previous experiment, depth images of the back of young calves can be used to train a network able to identify them during the three subsequent weeks, further improving the capacity of deep learning algorithms to monitor animals from an early stage of life. Such algorithms can contribute to the advancement of animal traceability and infectious diseases control, ultimately improving farm productivity, food safety, consumer trust, and production sustainability (Awad, 2016).

Deep learning algorithms can be used to identify individual animals using their dorsal area 3D surfaces and, based on our experiments using calves in their growing period, they are robust enough to account for changes in body shape and size of the same animals. This study focused on

calves in their early stage of life because that is when they undergo the most significant changes in body shape and size, representing a more challenging setting for machine learning algorithms, as significant divergence between training and future (or testing) data distributions often hinders such algorithms' predictive performance. Conversely, when working with mature cows that show a more limited body shape variability, the training data distribution would be more similar to that of images collected in the future (images of interest for identification), thus representing a less challenging setting for machine learning algorithms. In fact, Andrew et al. (2016) and Okura et al. (2019) used Red, Green, Blue, Depth (**RGB-D**) images to identify mature Holstein dairy cows. Nevertheless, adult dairy cows can still undergo significant body shape changes during the transition period (between late pregnancy and early lactation), as they mobilize fat stores to compensate for a high milk yield and relatively low dry matter intake. Thus, although not explicitly shown in this study, algorithms that are able to identify individual calves as their body shapes change have the potential to be useful for monitoring mature dairy cows during their transition period, and future studies could explore this possibility.

Depending on the task complexity, 2D CNNs with a higher representation capacity, as a consequence of having a greater number of trainable parameters, can achieve better results than their 3D counterparts on identifying individual animals as their bodies grow. Nevertheless, regardless of the representation approach, 3D information can be used in computer vision systems that identify individual animals based exclusively on their shapes, instead of relying on coat color pattern information. Methods that rely on unique color patterns, such as the ones proposed by Andrew et al. (2017), Bello et al. (2020), Yao et al. (2019), and Hansen et al. (2018), are limited to only certain animal breeds, in scenarios with no significant body occlusion. Alternatively, deep learning methods that use solely 3D information for individual identification can potentially be

applied on species and breeds that share similar color patterns across individuals, such as Jersey, Brown Swiss, and Angus cattle, Rambouillet sheep, Saanen goats, Yorkshire pigs, and others; and in production systems where animals can be covered in mud or dirt, such as free range systems for pigs. By enabling the use of animal biometrics to perform individual identification in a multitude of species, breeds and production systems, these 3D deep learning algorithms push the boundaries of animal traceability and phenotyping. Although Yukun et al. (2019) makes use of depth and RGB images to perform animal identification, to the best of our knowledge this is the first work to propose the exclusive use of depth images and 3D representations for individual animal identification through 2D and 3D CNNs. Furthermore, this is also the first study to evaluate the ability of convolutional neural networks to identify animals as they grow rapidly and experience intense body shape changes. Moreover, since they are based on animal biometrics that cannot be easily manipulated by humans, the methods proposed in this work provide a secure and automated way of tracking individual animals along the food supply chain, contributing as an additional tool for ensuring food safety to consumers.

As previously mentioned, although deep learning methods represent the state-of-the-art in many computer vision applications, they often require large amounts of training data to efficiently learn a certain task. This could pose an obstacle to commercial applications where labeled data is not so readily available. In that context, implementing hybrid approaches that merge traditional computer vision techniques with deep learning might help reduce the need for labeled data and decrease training times (O'Mahony et al., 2020). Alternatively, active learning techniques (Settles, 2009) can be used to include human input in the learning process to optimize data annotation (for example, the system could request more examples of cows that are harder to classify or classes that are underrepresented). In addition, semi-supervised learning methods can leverage

information contained in both labeled and unlabeled data to build high-performing classifiers, potentially requiring smaller amounts of labeled data for training (Zhu, 2005).

In future research, capturing images during longer periods of time throughout the animals' life might help understand how long a trained network can still be useful for individual recognition without the need to retrain it. Additionally, it would be interesting to explore how the proposed methods would apply to mature dairy cows and other animal species and breeds such as Angus cows, or Yorkshire pigs. It would be beneficial to include more individuals in a future study as well, bringing the context closer to that of a commercial farm. In fact, commercial farms rarely hold a fixed herd for a long time. Instead, animals are constantly added or removed from the herd, making it necessary to either retrain the algorithms to include new individuals, or use an approach that is more suitable for an open herd setting, such as the one described by Andrew et al. (2021), that used deep metric learning to identify cattle that have never been seen before by the network. However, this problem still needs to be addressed in a larger scale in order to effectively implement visual identification systems in commercial farms or whole production systems, where hundreds or even thousands of individuals need to be identified and monitored simultaneously. Future applications should be able to dynamically integrate new animals to the system as they are added to the herd, using mechanisms to tag images of never-before-seen animals for later identification. Potential approaches for addressing such problems are explored in the fields of self-supervised and zero-shot learning. Certain self-supervised learning techniques such as Contrastive Learning allow neural networks to extract semantic representations from high-dimensional data using unlabeled datasets (Le-Khac et al., 2020), which could then be used to identify and cluster new examples of individuals that had not been seen before by the system, creating a temporary label that could later be mapped to a cow identification number. Furthermore, zero-shot learning consists of classifying

samples that belong to classes not observed during training, given some auxiliary information, and recently proposed methods have proven successful in areas of research regarding computer vision jointly with natural language processing (Xian et al., 2019).

## CONCLUSION

The outcomes of this study show that it is possible to use computer vision systems to identify individual animals using the 3D surface of their dorsal body region. Both 2D and 3D representations of the dorsal surface, and the corresponding neural network architectures, can be used in such systems, each being more appropriate for different scenarios. Additionally, the experiments using images of calves taken during a period of intense growth provided evidence that neural networks can learn unique biometric features from the back of these animals, which remain recognizable even as body size changes. These findings suggest that it is possible to use neural networks to monitor and identify animals from an early stage of life, or as they experience rapid body changes. By using exclusively the body shape of the animals (either through 2D depth images or 3D voxels and point clouds), the proposed methods may potentially be applied to species and breeds from which individuals share similar coat color patterns, which would be impossible to recognize using RGB images. This contributes to a broader application of animal traceability and integrated phenotyping based on computer vision, facilitating infectious disease control, and improving farm productivity, food safety, consumer trust, and production sustainability. Future research can be developed towards investigating techniques such as semi-supervised and active learning, as well as hybrid approaches that merge traditional computer vision methods and deep learning, to provide systems that are more data efficient and potentially perform better when exposed to large amounts of unlabeled data. Additionally, future work pertaining to computer vision-based identification systems in large commercial farms should evaluate the potential of

novel self-supervised, zero-shot learning, and other techniques to overcome the challenge concerning dynamically changing herds.

## ACKNOWLEDGMENTS

This research was performed using the computational resources and assistance of the University of Wisconsin-Madison Center for High Throughput Computing (**CHTC**) in the Department of Computer Sciences. The CHTC is supported by University of Wisconsin-Madison, the Advanced Computing Initiative, the Wisconsin Alumni Research Foundation, the Wisconsin Institutes for Discovery, and the National Science Foundation, and is an active member of the Open Science Grid, which is supported by the National Science Foundation and the U.S. Department of Energy's Office of Science. The authors would like to thank the financial support from the USDA National Institute of Food and Agriculture (Washington, DC; grant 2020-67015-30831) and USDA Hatch (WIS03085).

## REFERENCES

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. 2016. TensorFlow: A System for Large-Scale Machine Learning. Pages 265–283 in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). USENIX Association, Savannah, GA.

Aijazi, A.K., P. Checchin, and L. Trassoudaine. 2013. Segmentation Based Classification of 3D Urban Point Clouds: A Super-Voxel Based Approach with Evaluation. Remote Sens (Basel) 5:1624–1650. doi:10.3390/rs5041624.

Andrew, W., J. Gao, S. Mullan, N. Campbell, A.W. Dowsey, and T. Burghardt. 2021. Visual identification of individual Holstein-Friesian cattle via deep metric learning. Comput Electron Agric 185:106133. doi:https://doi.org/10.1016/j.compag.2021.106133.

Andrew, W., C. Greatwood, and T. Burghardt. 2017. Visual Localisation and Individual Identification of Holstein Friesian Cattle via Deep Learning. Pages 2850–2859 in 2017 IEEE International Conference on Computer Vision Workshops (ICCVW).

Andrew, W., S. Hannuna, N. Campbell, and T. Burghardt. 2016. Automatic individual holstein friesian cattle identification via selective local coat pattern matching in RGB-D imagery. Pages 484–488 in 2016 IEEE International Conference on Image Processing (ICIP).

Awad, A.I. 2016a. From classical methods to animal biometrics: A review on cattle identification and tracking. Comput Electron Agric 123:423–435. doi:https://doi.org/10.1016/j.compag.2016.03.014.

Awad, A.I. 2016b. From classical methods to animal biometrics: A review on cattle identification and tracking. Comput Electron Agric 123:423–435. doi:https://doi.org/10.1016/j.compag.2016.03.014.

Bello, R.-W., A.Z. Talib, A.S.A. Mohamed, D.A. Olubummo, and F.N. Otobo. 2020. Image-based individual cow recognition using body patterns. Image (IN) 11:92–98.

Casino, F., T.K. Dasaklis, and C. Patsakis. 2019. A systematic literature review of blockchain-based applications: Current status, classification and open issues. Telematics and Informatics 36:55–81. doi:https://doi.org/10.1016/j.tele.2018.11.006.

Cheng, S., Z. Leng, E.D. Cubuk, B. Zoph, C. Bai, J. Ngiam, Y. Song, B. Caine, V. Vasudevan, C. Li, Q. V Le, J. Shlens, and D. Anguelov. 2020. Improving 3D Object Detection Through Progressive Population Based Augmentation. Pages 279–294 in Computer Vision – ECCV 2020. Springer International Publishing, Cham.

Cho, Y., and K. Yoon. 2014. An overview of calf diarrhea - infectious etiology, diagnosis, and intervention. J Vet Sci 15:1–17. doi:10.4142/jvs.2014.15.1.1.

Chollet, F. 2015. Keras.

Chollet, F. 2017. Xception: Deep learning with depthwise separable convolutions. Pages 1251–1258 in Proceedings of the IEEE conference on computer vision and pattern recognition.

Cominotte, A., A.F.A. Fernandes, J.R.R. Dorea, G.J.M. Rosa, M.M. Ladeira, E.H.C.B. van Cleef, G.L. Pereira, W.A. Baldassini, and O.R. Machado Neto. 2020. Automated computer vision system to predict body weight and average daily gain in beef cattle during growing and finishing phases. Livest Sci 232:103904. doi:https://doi.org/10.1016/j.livsci.2019.103904.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. Pages 248–255 in 2009 IEEE Conference on Computer Vision and Pattern Recognition.

Dieleman, S., J. Schlüter, C. Raffel, E. Olson, S.K. Sønderby, D. Nouri, D. Maturana, M. Thoma, E. Battenberg, and J. Kelly. 2015. Lasagne: first release. Zenodo: Geneva, Switzerland 3:74.

Dórea, J.R.R., E.A. French, and L.E. Armentano. 2017. Use of milk fatty acids to estimate plasma nonesterified fatty acid concentrations as an indicator of animal energy balance. J Dairy Sci 100:6164–6176. doi:https://doi.org/10.3168/jds.2016-12466.

Esslemont, R.J., and M.A. Kossaibati. 1999. The Cost of respiratory diseases in dairy heifer calves. Bov Pract (Stillwater) 33:174–178. doi:10.21423/bovine-vol33no2p174-178.

Fernandes, A.F.A., J.R.R. Dórea, and G.J. de M. Rosa. 2020. Image Analysis and Computer Vision Applications in Animal Sciences: An Overview. Front Vet Sci 7.

Gezawa, A.S., Y. Zhang, Q. Wang, and L. Yunqi. 2020. A Review on Deep Learning Approaches for 3D Data Representations in Retrieval and Classifications. IEEE Access 8:57566–57593. doi:10.1109/ACCESS.2020.2982196.

Hahner, M., D. Dai, A. Liniger, and L. Van Gool. 2020. Quantifying Data Augmentation for LiDAR based 3D Object Detection. arXiv e-prints arXiv:2004.01643. doi:10.48550/arXiv.2004.01643.

Hansen, M.F., M.L. Smith, L.N. Smith, M.G. Salter, E.M. Baxter, M. Farish, and B. Grieve. 2018. Towards on-farm pig face recognition using convolutional neural networks. Comput Ind 98:145–152. doi:https://doi.org/10.1016/j.compind.2018.02.016.

He, K., G. Gkioxari, P. Dollar, and R. Girshick. 2017. Mask R-CNN. Page in Proceedings of the IEEE International Conference on Computer Vision (ICCV).

Hinton, G., N. Srivastava, and K. Swersky. 2012. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. Cited on 14:2.

Kaneene, J.B., and H. Scott Hurd. 1990. The national animal health monitoring system in Michigan. III. Cost estimates of selected dairy cattle diseases. Prev Vet Med 8:127–140. doi:https://doi.org/10.1016/0167-5877(90)90006-4.

Kingma, D.P., and J. Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

LeCun, Y., Y. Bengio, and G. Hinton. 2015. Deep learning. Nature 521:436–444. doi:10.1038/nature14539.

Le-Khac, P.H., G. Healy, and A.F. Smeaton. 2020. Contrastive Representation Learning: A Framework and Review. IEEE Access 8:193907–193934. doi:10.1109/ACCESS.2020.3031549.

Lin, M., Q. Chen, and S. Yan. 2013. Network In Network. arXiv e-prints arXiv:1312.4400. doi:10.48550/arXiv.1312.4400.

Marcé, C., R. Guatteo, N. Bareille, and C. Fourichon. 2010. Dairy calf housing systems across Europe and risk for calf infectious diseases. Animal 4:1588–1596. doi:https://doi.org/10.1017/S1751731110000650.

Maturana, D., and S. Scherer. 2015. VoxNet: A 3D Convolutional Neural Network for real-time object recognition. Pages 922–928 in 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

Nair, V., and G.E. Hinton. 2010. Rectified linear units improve Restricted Boltzmann machines. Pages 807–814 in ICML 2010 - Proceedings, 27th International Conference on Machine Learning.

Narayanan, P.J., P.W. Rander, and T. Kanade. 1998. Constructing virtual worlds using dense stereo. Pages 3–10 in Proceedings of the IEEE International Conference on Computer Vision.

Okura, F., S. Ikuma, Y. Makihara, D. Muramatsu, K. Nakada, and Y. Yagi. 2019. RGB-D video-based individual identification of dairy cows using gait and texture analyses. Comput Electron Agric 165:104944. doi:https://doi.org/10.1016/j.compag.2019.104944.

O'Mahony, N., S. Campbell, A. Carvalho, S. Harapanahalli, G.V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh. 2020. Deep Learning vs. Traditional Computer Vision. Pages 128–144 in Advances in Computer Vision. Springer International Publishing, Cham.

Park, U., Y. Tong, and A.K. Jain. 2010. Age-invariant face recognition. IEEE Trans Pattern Anal Mach Intell 32:947–954. doi:10.1109/TPAMI.2010.14.

Perez, L., and J. Wang. 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. arXiv e-prints arXiv:1712.04621. doi:10.48550/arXiv.1712.04621.

Pini, S., G. Borghi, R. Vezzani, D. Maltoni, and R. Cucchiara. 2021. A Systematic Comparison of Depth Map Representations for Face Recognition. Sensors 21. doi:10.3390/s21030944.

Qi, C.R., H. Su, K. Mo, and L.J. Guibas. 2016. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. arXiv e-prints arXiv:1612.00593. doi:10.48550/arXiv.1612.00593.

Qian, N. 1999. On the momentum term in gradient descent learning algorithms. Neural Networks 12:145–151. doi:https://doi.org/10.1016/S0893-6080(98)00116-6.

Robbins, H., and S. Monro. 1951. A Stochastic Approximation Method. The Annals of Mathematical Statistics 22:400–407. doi:10.1214/aoms/1177729586.

Rousseeuw, P.J., and C. Croux. 1993. Alternatives to the median absolute deviation. J Am Stat Assoc 88:1273–1283. doi:10.1080/01621459.1993.10476408.

Seitz, S.M., B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. 2006. A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms. Pages 519–528 in 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06).

Settles, B. 2009. Active learning literature survey.

Simonyan, K., and A. Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv e-prints arXiv:1409.1556. doi:10.48550/arXiv.1409.1556.

Soilán Rodríguez, M., R. Lindenbergh, B. Riveiro Rodríguez, and A. Sánchez Rodríguez. 2019. Pointnet for the automatic classification of aerial point clouds. ISPRS Annals of Photogrammetry Remote Sensing and Spatial Information Sciences.

Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. 2015. Going Deeper With Convolutions. Page in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Szegedy, C., V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. 2016. Rethinking the Inception Architecture for Computer Vision. Page in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

The Theano Development Team, R. Al-Rfou, G. Alain, A. Almahairi, C. Angermueller, D. Bahdanau, N. Ballas, F. Bastien, J. Bayer, A. Belikov, A. Belopolsky, Y. Bengio, A. Bergeron, J. Bergstra, V. Bisson, J. Bleecher Snyder, N. Bouchard, N. Boulanger-Lewandowski, X. Bouthillier, A. de Brébisson, O. Breuleux, P.-L. Carrier, K. Cho, J. Chorowski, P. Christiano, T. Cooijmans, M.-A. Côté, M. Côté, A. Courville, Y.N. Dauphin, O. Delalleau, J. Demouth, G. Desjardins, S. Dieleman, L. Dinh, M. Ducoffe, V. Dumoulin, S. Ebrahimi Kahou, D. Erhan, Z. Fan, O. Firat, M. Germain, X. Glorot, I. Goodfellow, M. Graham, C. Gulcehre, P. Hamel, I. Harlouchet, J.-P. Heng, B. Hidasi, S. Honari, A. Jain, S. Jean, K. Jia, M. Korobov, V. Kulkarni, A. Lamb, P. Lamblin, E. Larsen, C. Laurent, S. Lee, S. Lefrancois, S. Lemieux, N. Léonard, Z. Lin, J.A. Livezey, C. Lorenz, J. Lowin, Q. Ma, P.-A. Manzagol, O. Mastropietro, R.T. McGibbon, R. Memisevic, B. van Merriënboer, V. Michalski, M. Mirza, A. Orlandi, C. Pal, R. Pascanu, M. Pezeshki, C. Raffel, D. Renshaw, M. Rocklin, A. Romero, M. Roth, P. Sadowski, J. Salvatier, F. Savard, J. Schlüter, J. Schulman, G. Schwartz, I. Vlad Serban, D. Serdyuk, S. Shabanian, É. Simon, S. Spieckermann, S. Ramana Subramanyam, J. Sygnowski, J. Tanguay, G. van Tulder, J. Turian, S. Urban, P. Vincent, F. Visin, H. de Vries, D. Warde-Farley, D.J. Webb, M. Willson, K. Xu, L. Xue, L. Yao, S. Zhang, and Y. Zhang. 2016. Theano: A Python framework for fast computation of mathematical expressions. arXiv e-prints arXiv:1605.02688. doi:10.48550/arXiv.1605.02688.

Voulodimos, A., N. Doulamis, A. Doulamis, and E. Protopapadakis. 2018. Deep learning for computer vision: A brief review. Comput Intell Neurosci 2018.

Voulodimos, A.S., C.Z. Patrikakis, A.B. Sideridis, V.A. Ntafis, and E.M. Xylouri. 2010. A complete farm management system based on animal identification using RFID technology. Comput Electron Agric 70:380–388. doi:https://doi.org/10.1016/j.compag.2009.07.009.

Wang, P., W. Solorzano, T. Diaz, C.E. Magyar, S.M. Henning, and J. V Vadgama. 2017. Arctigenin inhibits prostate tumor cell growth in vitro and in vivo. Clin Nutr Exp 13:1–11. doi:https://doi.org/10.1016/j.yclnex.2017.04.001.

Weiss, K., T.M. Khoshgoftaar, and D.D. Wang. 2016. A survey of transfer learning. J Big Data 3. doi:10.1186/s40537-016-0043-6.

Wu, Z., S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 2015. 3D ShapeNets: A Deep Representation for Volumetric Shapes. Page in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Xian, Y., C.H. Lampert, B. Schiele, and Z. Akata. 2019. Zero-Shot Learning—A Comprehensive Evaluation of the Good, the Bad and the Ugly. IEEE Trans Pattern Anal Mach Intell 41:2251–2265. doi:10.1109/TPAMI.2018.2857768.

Yao, L., Z. Hu, C. Liu, H. Liu, Y. Kuang, and Y. Gao. 2019. Cow face detection and recognition based on automatic feature extraction algorithm. Page in Proceedings of the ACM Turing Celebration Conference - China. Association for Computing Machinery, New York, NY, USA.

Yukun, S., H. Pengju, W. Yujie, C. Ziqi, L. Yang, D. Baisheng, L. Runze, and Z. Yonggen. 2019. Automatic monitoring system for individual dairy cows based on a deep learning framework that provides identification via body parts and estimation of body condition score. J Dairy Sci 102:10140–10151. doi:https://doi.org/10.3168/jds.2018-16164.

Zhu, X.J. 2005. Semi-supervised learning literature survey.

## TABLES AND FIGURES

**Table 2.1.** Splits performed for the second objective, experiment 2. The ten splits were grouped according to the number of weeks used for training, and the four resulting groups were compared to evaluate the effect of adding more weeks to the training set.

| Group | Weeks in training set | Test week | Test set size |
|---|---|---|---|
| Two weeks | 1 and 2 | 3 | 319 |
| | 2 and 3 | 4 | 254 |
| | 3 and 4 | 5 | 250 |
| | 4 and 5 | 6 | 403 |
| Three weeks | 1, 2, and 3 | 4 | 254 |
| | 2, 3, and 4 | 5 | 250 |
| | 3, 4, and 5 | 6 | 403 |
| Four weeks | 1, 2, 3, and 4 | 5 | 250 |
| | 2, 3, 4, and 5 | 6 | 403 |
| Five weeks | 1, 2, 3, 4, and 5 | 6 | 403 |

**Table 2.2.** Splits performed for the second objective, experiment 3. The ten splits were grouped according to the time interval between training and test sets, and the four resulting groups were compared to evaluate the effect of skipping weeks between training and testing.

| Group | Weeks in training set | Test week | Test set size |
|---|---|---|---|
| No skipping | 1 and 2 | 3 | 319 |
| | 2 and 3 | 4 | 254 |
| | 3 and 4 | 5 | 250 |
| | 4 and 5 | 6 | 403 |
| Skipping one week | 1 and 2 | 4 | 254 |
| | 2 and 3 | 5 | 250 |
| | 3 and 4 | 6 | 403 |
| Skipping two weeks | 1 and 2 | 5 | 250 |
| | 2 and 3 | 6 | 403 |
| Skipping three weeks | 1 and 2 | 6 | 403 |

**Table 2.3.** Experiments performed for the second objective. The experiments evaluated how changing the number of images per animal, number of weeks used for training, and time interval between training and testing affected the predictive performance of the algorithms.

| Experiment | Images per animal | Number of weeks | Time interval |
|---|---|---|---|
| 1 | Varying | 2 | No skipping |
| 2 | 80 | Varying | No skipping |
| 3 | 80 | 2 | Varying |

**Table 2.4.** $F_1$ scores for each combination of train-test split, data representation, and network architecture for objective 1. The best performing network was the one based on the Xception 2D CNN architecture.

| Train-test split | Data representation | Architecture | $F_1$ score |
|---|---|---|---|
| RO[1] | DI[3] | VGG16 | 0.888 |
| RO[1] | DI[3] | Inception v3 | 0.904 |
| **RO[1]** | **DI[3]** | **Xception** | **0.959** |
| RO[1] | PC[4] | PointNet | 0.669 |
| RO[1] | OG[5] | VoxNet | 0.880 |
| CO[2] | DI[3] | VGG16 | 0.718 |
| CO[2] | DI[3] | Inception v3 | 0.750 |
| **CO[2]** | **DI[3]** | **Xception** | **0.804** |
| CO[2] | PC[4] | PointNet | 0.429 |
| CO[2] | OG[5] | VoxNet | 0.656 |

[1]RO = Random order.
[2]CO = Chronological order.
[3]DI = Depth images.
[4]PC = Point cloud.
[5]OG = Occupancy grid (voxel).

**Table 2.5.** $F_1$ scores for each combination of images per animal and network architecture for the first experiment of objective 2. The VoxNet-based network achieved the best results in this experiment. Increasing the number of training images generally improved the $F_1$ scores, up to around 100 images per animal.

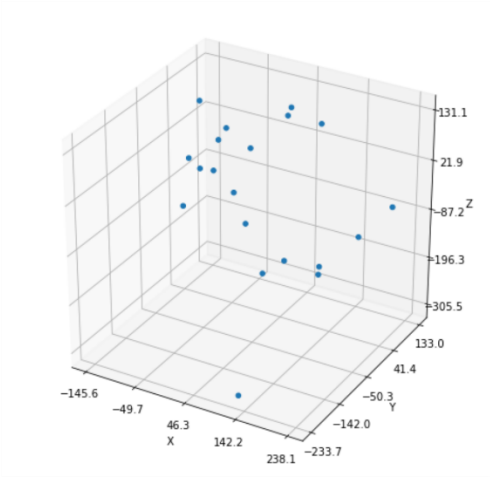
| Images per animal | VGG16 | Inception v3 | Xception | PointNet | VoxNet |
|---|---|---|---|---|---|
| 20 | 0.641 | 0.656 | 0.539 | 0.603 | **0.734** |
| 40 | 0.546 | 0.770 | 0.701 | 0.697 | **0.917** |
| 70 | 0.558 | 0.859 | 0.827 | 0.656 | **0.929** |
| 100 | 0.605 | 0.757 | 0.852 | 0.727 | **0.944** |
| 130 | 0.629 | 0.788 | 0.910 | 0.653 | **0.947** |
| 154 | 0.643 | 0.763 | 0.858 | 0.630 | **0.939** |

**Table 2.6.** $F_1$ scores for each combination of number of weeks used for training and network architecture for the second experiment of objective 2. The Xception-based network achieved the best results in this experiment. The highest score improvement happened when adding a fourth week to the training set.

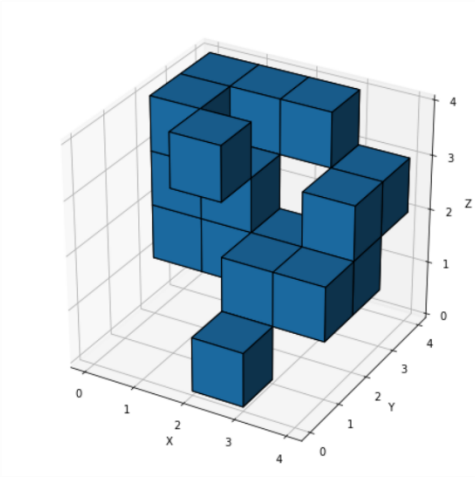
| Number of weeks | VGG16 | Inception v3 | Xception | PointNet | VoxNet |
|---|---|---|---|---|---|
| 2 | 0.683 | 0.795 | 0.909 | 0.643 | **0.911** |
| 3 | 0.724 | 0.776 | **0.906** | 0.581 | 0.880 |
| 4 | 0.695 | 0.706 | **0.970** | 0.463 | 0.903 |
| 5 | 0.747 | 0.635 | **0.979** | 0.395 | 0.888 |

**Table 2.7.** $F_1$ scores for each combination of number of weeks skipped between training and testing and network architecture, for the third experiment of objective 2. The Xception-based network achieved the best results in this experiment. Skipping one week affected the $F_1$ score, but it remained roughly constant after further skipping more weeks.

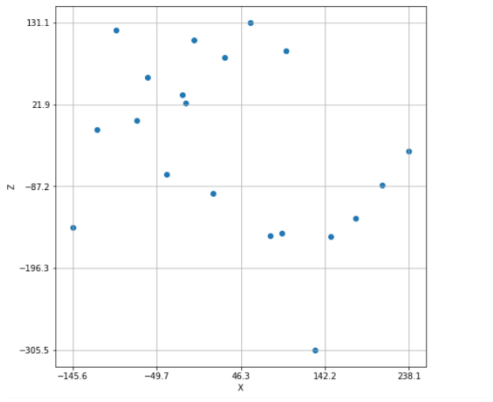
| Time interval | VGG16 | Inception v3 | Xception | PointNet | VoxNet |
|---|---|---|---|---|---|
| No skipping | 0.704 | 0.746 | **0.917** | 0.533 | 0.917 |
| 1 week | 0.595 | 0.612 | **0.846** | 0.551 | 0.831 |
| 2 weeks | 0.535 | 0.654 | **0.835** | 0.441 | 0.806 |
| 3 weeks | 0.753 | 0.726 | **0.856** | 0.282 | 0.792 |



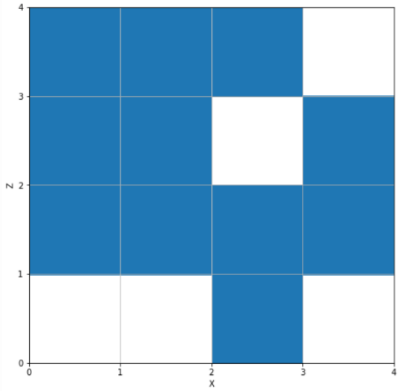

(a) Point cloud in 3D space




(b) Occupancy grid generated in a 4×4×4 grid space

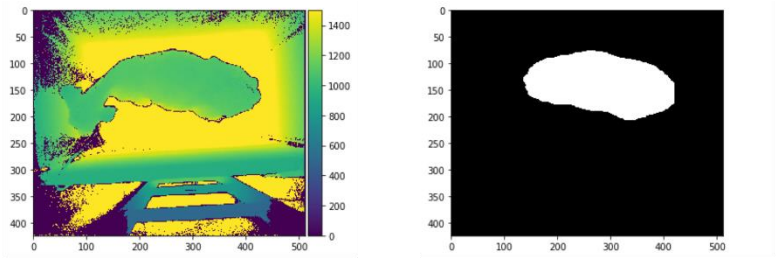



(c) Point cloud projected on the XZ-plane

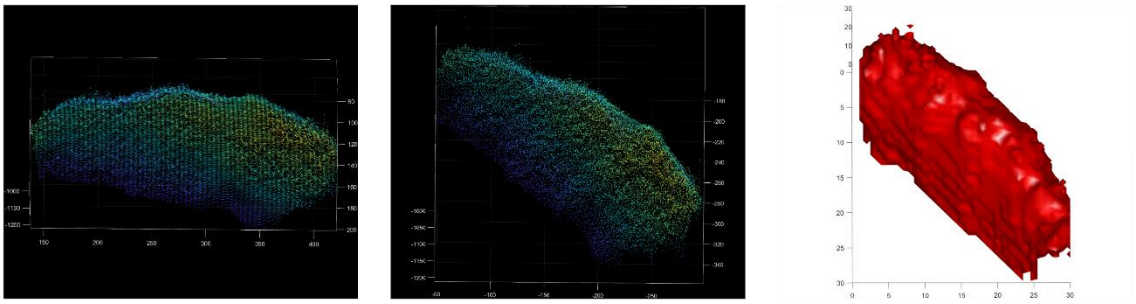



(d) Occupancy grid projected on the XZ-plane

**Figure 2.1.** Example of the occupancy grid generation process: (a) shows a point cloud in 3D space, (b) shows the corresponding generated occupancy grid in a 4 × 4 × 4 grid space, (c) shows the same point cloud projected onto the XZ-plane, and (d) shows the corresponding occupancy grid projected onto the XZ-plane. In the occupancy grids (b and d), filled cells are assigned value 1, and empty cells are assigned value 0. Examples were given in both 3D and 2D for clarification.

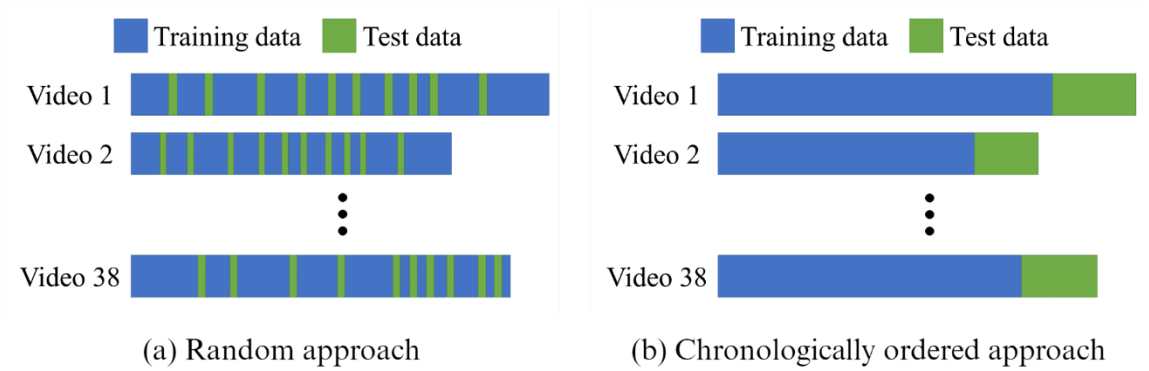


(a) Original depth frame (b) Output from Mask R-CNN



(c) Generated point cloud (d) Augmented point cloud (e) Generated occupancy grid

**Figure 2.2.** An example of all preprocessing stages applied to a depth image to generate an occupancy grid. A depth frame (a) is extracted from a video captured using the Kinect V2 sensor; a Mask R-CNN network detects the pixels containing the calf body, and generates a binary mask (b); this binary mask is applied to the point cloud generated from the depth frame, resulting in a point cloud of the calf body (c); this point cloud is then augmented (d) and used to generate the final occupancy grid (e).




(a) Random approach (b) Chronologically ordered approach

**Figure 2.3.** Dataset splits for the first objective. In the random approach (a), the dataset was randomly split into training and test sets, including 80% and 20% of the frames, respectively. In the chronologically ordered approach (b), the frames from each video were assigned to the training or test sets based on their positions in the video: the first 80% frames were assigned to the training set, and the last 20% were assigned to the test set.

# CHAPTER THREE: USING PSEUDO-LABELING TO IMPROVE PERFORMANCE OF DEEP NEURAL NETWORKS FOR ANIMAL IDENTIFICATION

## ABSTRACT

Contemporary approaches for animal identification use deep learning techniques to recognize coat color patterns and identify individual animals in a herd. However, deep learning algorithms usually require a large number of labeled images to achieve satisfactory performance, which creates the need to manually label all images when automated methods are not available. In this study, we evaluated the potential of a semi-supervised learning technique called pseudo-labeling to improve the predictive performance of deep neural networks trained to identify Holstein cows using labeled training sets of varied sizes and a larger unlabeled dataset. By using such technique to automatically label previously unlabeled images, we observed an increase in accuracy of up to 20.4 percentage points compared to using only manually labeled images for training. Our final best model achieved an accuracy of 92.7% on an independent testing set to correctly identify individuals in a herd of 59 cows. These results indicate that it is possible to achieve better performing deep neural networks by using images that are automatically labeled based on a small dataset of manually labeled images using a relatively simple technique. Such strategy can save time and resources that would otherwise be used for labeling, and leverage well annotated small datasets.

## INTRODUCTION

Computer vision systems (**CVS**) have great potential to generate precise high-throughput phenotyping in several domains, such as precision medicine, crop and animal breeding, and farm

management. Deep neural network algorithms are the state-of-the-art in such computer vision tasks and they often require large amounts of data to achieve satisfactory performance (LeCun et al., 2015). Supervised learning tasks require the training data to be annotated and most image data generated by CVS in agriculture are not automatically annotated or easy to annotate. Additionally, such CVS have the potential to generate large amounts of data that are labor- and resource-intensive to organize, annotate, and analyze. Objects of interest in agriculture setups are usually challenging to be manually annotated by humans, such as individual crops in a farm plot or individual animals in a herd, resulting in an even more laborious and time-consuming annotation process which is often prone to human error. Several techniques have been proposed in the past decades attempting to enable deep neural networks to learn from small datasets, reducing costs related to data collection, annotation, and preprocessing while maintaining good predictive performance. Such trend can be noticed through the transition from strictly supervised approaches with large, annotated datasets to approaches that use partially annotated or unlabeled data that require less or no annotation. Among those techniques, there have been great advances in the field of few-shot learning (Wang et al., 2020), and more notably semi-supervised learning (**SSL**) (van Engelen and Hoos, 2020).

In livestock systems, animal identification is the first step for individual animal phenotyping. Current state-of-the-art computer vision methods for animal identification usually require large labeled datasets that can be labor-intensive to annotate. Andrew et al. (2017) and (Xiao et al., 2022) trained convolutional neural networks (**CNN**) to identify Holstein cows using top-down view images of their back, Yao et al. (2019) used detection and classification CNNs for face detection and recognition of individual Holstein cows, Yukun et al. (2019) used Red, Green, Blue, Depth (**RGB-D**) images to identify Holstein cows and estimate their body condition scores,

and Hansen et al. (2018) proposed their own 2-dimensional (**2D**) CNN to individually identify pigs using images of their faces. Although such studies focused on a closed-set problem, which involves identification within a fixed group of animals, this scenario is not commonly encountered in commercial farms where the animal movement in and out of the herd is highly dynamic. Consequently, Andrew et al. (2021) introduced a novel approach to address the challenge of identifying individuals in an open-set scenario. Their method utilizes deep metric learning to generate image representations that exist within a latent space that facilitates the clustering of images belonging to the same animal, allowing for individual identification even in dynamic environments. Nevertheless, it remains unclear how these proposed methods would effectively scale for application in large commercial herds. This includes not only evaluating their predictive performance but also addressing the operational challenges associated with data collection, processing, and annotation.

As previously mentioned, existing studies on animal identification have predominantly focused on fully supervised approaches that require extensive image annotation for both closed-set and open-set scenarios, which can be labor-intensive and time-consuming. In this context, semi-supervised learning can be an effective tool for leveraging unlabeled data collected from camera systems installed at farms that would otherwise require significant human effort to annotate. In SSL, the machine learning algorithm learns structured information from the labeled portion of the dataset and uses the patterns captured from the unlabeled data to improve its predictive performance and generalization power. Thus, scenarios where labeling all the data available is too expensive or even unfeasible, but it is still possible to label part of the dataset, are the most adequate for SSL. In the context of livestock systems, Zhang et al. (2022) introduced an SSL method for teat-end condition classification on dairy cows and found a significant

improvement in performance by taking advantage of unlabeled data through their proposed algorithm. However, applications of SSL for individual animal identification using computer vision are yet to be explored.

Within SSL, a popular technique is pseudo-labeling, which consists of iteratively including confident predictions of unlabeled data into the training dataset (Lee, 2013). Pseudo-labeling allows for a simple and effective way to improve the predictive performance of trained machine learning models when labeling more data is costly and large amounts of unlabeled data are available. Pseudo-labeling can be easily implemented with various machine learning algorithms applied to different datasets (if unlabeled data is available) and domains, including applications in agriculture (Yao et al., 2016; Qiao et al., 2022), medicine (Momoki et al., 2022), person re-identification (Wu et al., 2018), and remote sensing (Zhou and Li, 2020), for example. The simplicity and versatility of pseudo-labeling were our main motivations for evaluating the application of this technique for training deep neural networks for animal identification.

The objective of this study was to evaluate the potential of a semi-supervised learning technique called pseudo-labeling to improve the predictive performance of deep convolutional neural networks trained to identify individual Holstein cows using labeled training sets of varied sizes and a larger unlabeled dataset. The core emphasis of this work was not on introducing a novel SSL method, but rather to address a biological problem— the identification of individual animals, and to present a fresh perspective to approach this issue—that of semi-supervised learning. Thus, we focused on studying pseudo-labeling in the novel setting of animal identification, rather than proposing extensive modifications to current semi-supervised methods. The method evaluated in this study is complementary to current animal identification research, as it can be seamlessly

applied to previously trained models without requiring any modifications in the model architecture or optimization procedure.

## MATERIAL AND METHODS

### Data Collection

Images from 59 lactating cows were taken using four Intel RealSense D435 depth cameras (Keselman et al., 2017) installed at the milking parlor exit lanes of the Emmons Blaine Dairy Cattle Research Center (Arlington, WI) between August 8th and October 7th, 2020. Top-down view images were captured twice a day following each milking session, triggered by cow presence detection within camera range. Because the cameras contained a depth sensor, the method for detecting a cow under the camera consisted of checking if a region inside the lane had an average distance from the camera below a certain threshold. For this study, we used a threshold value of 3 m, meaning that a snapshot would be taken only if there was an object less than 3 m away from the camera. Given that the cameras were installed at 3.5 m high, snapshots were taken if and only if there was a cow under the camera. In total, 23,709 snapshots were used in this study, of which 4,695 were labeled with the corresponding cow identification code, and 20,194 were kept unlabeled. The labeled snapshots were split into training, validation, and test sets according to the capture date, as shown in Table 3.1. The validation set was used to define the best threshold values for each round of the pseudo-labeling algorithm (see details in *Pseudo-Labeling*), and the test set worked as a final independent performance assessment.

### Data Preprocessing

Each snapshot consisted of a depth and an infrared image, both with a resolution of 640 × 480 pixels. The depth image contained, for each pixel, the distance in millimeters from the object in that pixel to the camera sensor, and the infrared image contained a value between 0 and 255 for

each pixel, ranging from black to white, respectively. An image segmentation algorithm based on Mask R-CNN (He et al., 2017) with ResNet-50 (He et al., 2016) as the backbone architecture was trained using a dataset of 843 depth images and the corresponding manually defined segmentation masks of cows and calves in dairy farms. This trained cow segmentation algorithm was applied to the depth images to generate segmentation masks for each snapshot, which were then used to remove all background pixels from the infrared images (i.e., pixels that did not contain the cow's body, from tail to neck). Finally, the segmented images were cropped to include only the area containing the cow body and rotated to adjust the cow to a horizontal position. All further experiments used segmented infrared images. See Figure 3.1 for an example of the preprocessing steps applied to each snapshot.

Neural Network Training

All neural networks in this study were trained using the Keras (Chollet, 2015) library available in Python, with TensorFlow (Abadi et al., 2016) as the backend. For the full iteration of pseudo-labeling trained in four rounds (see details in *Pseudo-Labeling*), the neural networks followed the Xception architecture (Chollet, 2017). We have also evaluated two additional architectures for one round of pseudo-labeling—MobileNetV2 (Sandler et al., 2018) and NASNet Large (Zoph et al., 2018). We selected these architectures purposefully to represent a broader spectrum of the design philosophies in deep learning—MobileNetV2, a lightweight and efficient design suitable for mobile and embedded vision applications; NASNet Large, a modern and high-performing architecture developed through neural architecture search; and Xception, an architecture that uses depthwise separable convolutions to enhance model efficiency, and it represents a good trade-off between model complexity and predictive performance. We compared the performance of each neural network architecture after one round of pseudo-labeling to study

how such design decisions interact with the SSL approach. However, it is important to underline that our main objective was not to comprehensively compare deep learning architectures per se, but to better understand the impact of using pseudo-labeling for enhancing existing animal identification models.

All networks were trained with Transfer Learning using the ImageNet dataset (Deng et al., 2009), meaning that the weights from all layers except for the last two fully-connected (**FC**) layers were initialized with the values from the original corresponding networks trained using ImageNet. This technique accelerates the training process as it allows the neural network to retain the knowledge previously learned from training using a large generic image dataset.

During training, image augmentation was performed using the built-in image augmentation functionality from Keras, with the following parameters: *zoom_range* = 0.1, *brightness_range* = (0.2, 1.5), *horizontal_flip* = True, *vertical_flip* = True, *fill_mode* = 'nearest'. This means that, during the training procedure, all training images were randomly zoomed in or out by up to 10%, had the pixel brightness adjusted to a random value between 20 and 150% of the original, and had a 50% chance of being flipped horizontally or vertically. For each neural network, the training process was performed in two stages: feature extraction and fine-tuning. In the feature extraction stage, the neural network was trained for 30 epochs with only the weights from the last two FC layers unfrozen, keeping all other weights (i.e., the ones learned from ImageNet) unchanged. Then, in the fine-tuning stage, weights from earlier layers were unfrozen and the network was trained for 60 epochs with a smaller learning rate, allowing the network to adjust its weights to our more specific datasets and tasks. The weights were optimized using the Adam algorithm (Kingma and Ba, 2014) with a learning rate of $1 \times 10^{-3}$ in the feature extraction stage and $1 \times 10^{-5}$ in the fine-tuning stage.

Pseudo-Labeling

The technique explored in this study consists of training a convolutional neural network in multiple “rounds”, with each round consisting of the following steps: first, an initial training set labeled by humans is used to train a neural network for cow identification; then this trained network performs predictions on a larger unlabeled dataset; and finally, the unlabeled images with confident predictions are added to the training set containing previously labeled images for training a new neural network. A confidence threshold value controls which unlabeled images are included in the training set for the next round, such that only images with a prediction confidence above that threshold are included. Thus, the threshold value works as a trade-off between training set size and pseudo-label quality, dictating whether the next round will contain more images with uncertain predicted labels, or fewer but more certain image labels. The new neural network is trained using both the original manually labeled dataset and the portion of the unlabeled dataset for which the prediction probabilities were above the defined threshold. After that, another round of predictions is performed on the remaining unlabeled data, and the new images and corresponding predictions are included in the next pseudo-labeling round. This process is repeated until a given stopping condition is achieved. For this study we performed up to four rounds of pseudo-labeling for each experiment. Figure 3.2 illustrates the steps that compose one round of pseudo-labeling.

It is important to note the difference between the technique explored in this study and the one proposed by Lee (2013). Lee (2013) proposes that labeled and unlabeled data are used simultaneously during the training schedule and that the pseudo-labels are recalculated after every weight update. Alternatively, in this study we perform multiple rounds of training, including new unlabeled data with the corresponding predicted labels only after full training schedules. We chose this method so that we could define a threshold value after each training round to only include

unlabeled images with higher probabilities, as opposed to including every unlabeled image in the entire training procedure. Additionally, the method used in this study can be seamlessly applied to previously trained networks without any modifications in the original network architecture or optimization procedure, which might prove useful as a complementary, additional step for enhancing current animal identification networks. Part of our experiments consisted of finding the best threshold values based on the initial labeled training set.

## Experiments

We performed four types of experiments to evaluate the best scenarios for applying pseudo-labeling for animal identification using deep neural networks. Such experiments consisted of (1) varying the confidence threshold for a prediction to be included in the next training step, (2) evaluating different neural network architectures for one round of pseudo-labeling, (3) performing multiple rounds of pseudo-labeling for one of the architectures, and (4) evaluating the effectiveness of this technique with varying manually labeled initial training set sizes.

### Threshold Values

When performing pseudo-labeling, a confidence threshold value must be defined to dictate which unlabeled images and their corresponding predictions are included in the training set for the next training round. This confidence threshold is applied over the predicted confidence values generated by the trained neural network for each new image. The last layers of the deep neural networks utilized in this study contained a softmax activation function with the number of output units corresponding to the number of classes (in this case, one class for each animal, resulting in 59 units). This means that the softmax function, with the formula described in Eq. 3.1, was applied to the output of such networks, resulting in output values between 0 and 1 for each class, and the total sum of all output values equaling exactly 1. For that reason, the output of a neural network

that contains a softmax activation function in its last layer can be interpreted as the confidence value that the network believes a given data point belongs to each class.

$$\sigma(\boldsymbol{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad \text{(Eq. 3.1)}$$

where **z** is the *K*-dimensional output of the network before applying softmax, *i* and *j* are class indices, *e* is Euler's number, *K* is the total number of classes, and $\sigma$ represents the softmax function.

The threshold values evaluated in this study were 0 (meaning that all unlabeled data would be included in the next training set), 0.5, 0.75, 0.90, 0.95, 0.98, 0.99, 0.995, 0.999, 0.9999, 0.99999, 0.999999, and 0.9999999. These values were chosen based on an analysis of the output confidence values generated by the initial trained network on unlabeled images. The implementation of the neural networks in this study allowed for output values represented by a 64-bit floating-point variable in Python ranging from 0 to 1, which can accommodate real numbers with 7 decimal digits or more, thus allowing for all threshold values chosen to be relevant. It is important to note that optimal threshold values for pseudo-labeling are highly dependent on the machine learning algorithm used and its possible output values.

Different Neural Network Architectures

Aiming to study the impact of different architectural design philosophies on the use of pseudo-labeling for this particular problem, we evaluated three neural network architectures for one round of pseudo-labeling—Xception (Chollet, 2017), MobileNetV2 (Sandler et al., 2018), and NASNet Large (Zoph et al., 2018). We chose a variety of architectures that have varying number of parameters and design paradigms; however, it was not within the scope of this work to perform a comprehensive comparison of deep neural network architectures.

Multiple Rounds

For the Xception architecture, we evaluated the impact of performing multiple rounds of pseudo-labeling. We chose to further explore the technique on such architecture because it provided a good trade-off between predictive performance, number of parameters, and training time. For each pseudo-labeling round, the best threshold value was chosen based on the accuracy on the validation set, with potentially different optimal threshold values being selected on each round. After four rounds of pseudo-labeling, we evaluated the predictive performance of the final resulting Xception model on the test set and compared it with that achieved by the original model (trained using only manually labeled images) on the same test set.

Initial Labeled Training Set Size

In order to evaluate how the initial proportion of labeled and unlabeled images affects the final achieved accuracy after performing multiple rounds of pseudo-labeling, we generated random reduced versions of the initial manually labeled training set. Datasets containing 10%, 25%, 50%, 75%, and 90% of the manually labeled training set were generated using random sampling and used as initial training sets for four rounds of pseudo-labeling each. The generated datasets contained 235, 588, 1,177, 1,765, and 2,118 manually labeled images, respectively, which resulted in labeled proportions of 1%, 3%, 5%, 8%, and 9%. For this experiment, the Xception architecture was evaluated for four pseudo-labeling rounds.

Evaluation Metrics

Each threshold value generated a different neural network, trained using both the labeled images and the unlabeled images whose prediction confidences surpassed the confidence threshold. For each pseudo-labeling round, the best threshold value was the one that generated the neural network that achieved the highest accuracy on the validation set (consisting of 1,161 images). Both the baseline neural network, trained using just manually labeled images, and the

models generated after each pseudo-labeling round were evaluated on an independent test set containing 1,180 images that were not included in the training or the validation sets, in order to assess the performance improvements achieved from performing pseudo-labeling. For each model, both the accuracy and the Mean Average Precision (**mAP**) on the test set were calculated. Accuracy corresponds to the proportion of correctly classified images over the total number of images in the test set, and mAP corresponds to the micro-average of the area under the precision-recall curve for each class, averaged over all classes. Accuracy was used as the main performance metric because both validation and test sets were balanced, meaning that the number of images per class was the same for all classes, and mAP was calculated to allow for comparison with previous related work in dairy cattle identification.

## RESULTS AND DISCUSSION

The baseline Xception model, trained using the initial labeled training set containing all 2,354 images, achieved an accuracy of 83.45% on the validation set and 77.54% on the test set. As described in *Evaluation Metrics*, the calculated mAP was 90.48% on the validation and 85.23% on the testing set. Although the performance on the validation set was slightly higher than on the testing set, there is no direct explanation for that difference as the images from both validation and testing sets were taken on different days from those in the training set, as previously explained in *Data Collection*. The baseline MobileNetV2 and NASNet Large models achieved accuracies of 81.55% and 85.00% on the validation set, and 75.00% and 77.03% on the test set, respectively. These findings were consistent with our expectations, given the differences in model architecture and parameter count. MobileNetV2, being a lightweight model with 3.5 million parameters, NASNet Large, a larger model with 88.9 million parameters, and Xception, which falls in between

with 22.9 million parameters, demonstrated the well-known trend that models with higher parameter counts tend to exhibit greater predictive performance.

In our study, hyperparameter tuning was not performed for training the neural networks, since the network architectures were the same as in the corresponding papers, and the weights were initialized using Transfer Learning. Still, we generated a validation set to choose the best threshold values for each round of the pseudo-labeling algorithm. Because of that, we compared the predictive performance of the baseline models (trained using the 2,354 labeled training set images, before any pseudo-labeling was performed) on the validation set with other previously published studies using computer vision to identify Holstein cows. Zhao and He (2015) found an accuracy of 90.55% using side-view images of 30 cows; Andrew et al. (2017) found an mAP of 86.07% for individual identification among 89 cows; Zin et al. (2018) found an accuracy of 97.01% among 45 cows; and more recently, Xiao et al. (2022) achieved an accuracy of 98.67% using top-view images of 48 cows. These results indicate that the performance of our baseline models agreed with other similar studies that used computer vision to identify Holstein cows based on their coat color patterns.

On the first round of pseudo-labeling as described in *Pseudo-Labeling*, the total number of images used for training decreased almost exponentially as the threshold value approached 1. This phenomenon persisted across all three evaluated architectures, as shown in Figure 3.3. Lower threshold values add more images to the next training round, however, with more uncertainty on the pseudo-labels. Conversely, higher threshold values restrict the images used in the next training round to only those that contain pseudo-labels with higher confidence, decreasing the training set size but potentially increasing the quality of the pseudo-labels. As shown in Figure 3.4, the accuracy on the validation set starts increasing as the threshold increases, until it reaches a

maximum value at 0.999 (for Xception and MobileNetV2) or 0.99999 (for NASNet Large), and then starts decreasing as the threshold value increases further. These results reveal how one can adjust the threshold value to control the trade-off between the number of images used for training and the pseudo-label quality of the previously unlabeled images added after pseudo-labeling. Finding the threshold value that optimizes this trade-off is key for achieving the best results when using this pseudo-labeling technique.

For each architecture, we evaluated the best achieved test accuracy after one round of pseudo-labeling and the average training times on the same NVIDIA GeForce RTX 2080 Graphics Processing Unit (**GPU**) (NVIDIA; Santa Clara, CA). Results are shown in Table 3.2.

Since the Xception architecture provided the best trade-off between predictive and computational performance, we decided to only use this architecture for further experiments. For the Xception architecture, the best threshold value (i.e., the one that maximizes validation accuracy) in the first round of pseudo-labeling was found to be 0.999. The new model, trained using both the initial manually labeled training set and unlabeled images with a prediction confidence of above 0.999, was then used to perform predictions on the remaining unlabeled images, resulting in a new round of pseudo-labeling. On this second round, the best threshold value found was 0.999999, achieving a validation accuracy of 94.66%, as shown in Figure 3.5. After performing the same procedure two more times, the model resulting from the third round of pseudo-labeling used a total of 21,667 images for training (2,354 manually labeled and 19,313 pseudo-labeled), and the model resulting from the fourth round of pseudo-labeling used a total of 22,418 images for training (2,354 manually labeled and 20,064 pseudo-labeled). The final model achieved an accuracy of 95.25% on the validation set and 92.71% on the test set, consisting of a 15.17% absolute and 19.6% relative increase on testing accuracy when compared to the original

model trained using just the manually labeled images. These results show the great potential for improving the predictive performance of trained neural networks by using this relatively simple pseudo-labeling technique to leverage the information contained in large unlabeled image datasets.

Lee (2013) proposed the inclusion of an alpha hyperparameter which dictates the relative weight of the unlabeled portion of the data on the loss function value calculated at each training iteration. This process is equivalent to Entropy Regularization, and the choice of alpha controls the trade-off between giving more importance to unlabeled or labeled data during each stage of training. The rationale for defining a schedule for alpha is that in earlier epochs its value should be low, allowing the network to learn mostly from labeled data, and as the network becomes more proficient throughout training, the alpha value can be adjusted to higher values to allow for unlabeled data to be included in training with potentially more accurate predictions. This procedure uses every image from the unlabeled dataset, attributing a weight to the importance of the entire unlabeled dataset during training. Conversely, the confidence thresholding technique utilized in this study allows for a more discriminating choice of unlabeled images to be used in subsequent training rounds, completely excluding part of the unlabeled dataset, but simultaneously weighting labeled and unlabeled images the same during training.

Oliver et al. (2018) evaluated multiple SSL methods, focusing exclusively on those which consist of adding an additional loss term during training. The methods assessed in their study were either based on Consistency Regularization (Π-Model, Mean Teacher, and Virtual Adversarial Training), or pseudo-labeling. Their implementation of pseudo-labeling was like that proposed by Lee (2013) in the sense that unlabeled and labeled data were used in training simultaneously. However, Oliver et al. (2018) did not discuss the use of an alpha hyperparameter, and instead

incorporated thresholding, with a fixed value of 0.95 as found during their hyperparameter tuning procedure.

In the current study, similarly to Oliver et al. (2018), we used a thresholding parameter to select which unlabeled images would be used for training, however, their corresponding pseudo-labels were not updated dynamically during training. Instead, the pseudo-labels were only updated after each full round of training, including new unlabeled images as their corresponding prediction confidences reached a value above the defined threshold. Defining the range of thresholds to be tested during hyperparameter tuning required a careful evaluation of prediction confidences in the unlabeled dataset. Threshold values that differed only after the $5^{th}$ decimal place, for example, still resulted in significant performance and training set size differences, as seen in Figures 3.3 and 3.4. Figure 3.6 illustrates histograms of the confidence values predicted by the baseline fully supervised Xception model on the unlabeled dataset.

As described in *Initial Labeled Training Set Size*, the same 4-round procedure was repeated starting with reduced baseline datasets. The original manually labeled training set was reduced to 10%, 25%, 50%, 75%, and 90% of its size, maintaining class proportions. Then, the accuracy on the same fixed testing set was evaluated at the end of the fourth round and compared to the test accuracy before performing pseudo-labeling. The results are shown in Table 3.3. Even when reducing the starting training dataset to 50% of its original size, the network resulting after the end of four rounds of pseudo-labeling could achieve a predictive performance better than that of the full manually labeled dataset. Even on very small labeled datasets (average of 20 images per cow, which corresponds to approximately 5% of the total number of labeled and unlabeled images dedicated to training), performing this pseudo-labeling technique could still significantly improve the accuracy of the trained neural networks. In other words, by applying the pseudo-labeling

technique explored in this study, a neural network trained for animal identification using a fraction of the labeled images can achieve comparable or even better results than a neural network trained using more labeled images but without performing pseudo-labeling.

## CONCLUSION

The main goal of this study was to present a new perspective to approach the problem of animal identification using computer vision—using semi-supervised learning. We evaluated the potential of a relatively simple semi-supervised learning technique called pseudo-labeling to improve the predictive performance of neural networks trained to identify individual Holstein cows. The method evaluated in this study is complementary to current animal identification research, as it can be seamlessly applied to previously trained models without requiring modifications in the model architecture or optimization procedure. We believe that this use-inspired research highlights the potential of the evaluated method as a tool for advancing the field of animal identification, as it could be applicable for both closed- and open-set problems. Subsequent research could focus on comparing the evaluated method with other semi-supervised learning techniques, including those involving retraining a model from scratch. Furthermore, there is room for proposing modifications to existing SSL methods, tailoring the algorithms specifically to the task of animal identification. Additionally, it would be interesting to explore the efficacy of SSL techniques in the open-set scenario, as it reflects a more realistic setting for dynamic commercial herds.

## ACKNOWLEDGMENTS

This research was performed using the computational resources and assistance of the University of Wisconsin- Madison Center for High Throughput Computing (**CHTC**) in the Department of Computer Sciences. The CHTC is supported by University of Wisconsin-Madison,

the Advanced Computing Initiative, the Wisconsin Alumni Research Foundation, the Wisconsin Institutes for Discovery, and the National Science Foundation, and is an active member of the Open Science Grid, which is supported by the National Science Foundation and the U.S. Department of Energy's Office of Science. The authors would like to thank the financial support from the USDA National Institute of Food and Agriculture (Washington, DC; grant 2023-68014-39821/accession no. 1030367) and USDA Hatch (Accession number: 7002609).

## REFERENCES

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. 2016. TensorFlow: A System for Large-Scale Machine Learning. Pages 265–283 in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16). USENIX Association, Savannah, GA.

Andrew, W., J. Gao, S. Mullan, N. Campbell, A.W. Dowsey, and T. Burghardt. 2021. Visual identification of individual Holstein-Friesian cattle via deep metric learning. Comput Electron Agric 185:106133. doi:https://doi.org/10.1016/j.compag.2021.106133.

Andrew, W., C. Greatwood, and T. Burghardt. 2017. Visual Localisation and Individual Identification of Holstein Friesian Cattle via Deep Learning. Pages 2850–2859 in 2017 IEEE International Conference on Computer Vision Workshops (ICCVW).

Chollet, F. 2015. Keras.

Chollet, F. 2017. Xception: Deep learning with depthwise separable convolutions. Pages 1800–1807 in Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. Pages 248–255 in 2009 IEEE Conference on Computer Vision and Pattern Recognition.

van Engelen, J.E., and H.H. Hoos. 2020. A survey on semi-supervised learning. Mach Learn 109:373–440. doi:10.1007/s10994-019-05855-6.

Hansen, M.F., M.L. Smith, L.N. Smith, M.G. Salter, E.M. Baxter, M. Farish, and B. Grieve. 2018. Towards on-farm pig face recognition using convolutional neural networks. Comput Ind 98:145–152. doi:https://doi.org/10.1016/j.compind.2018.02.016.

He, K., G. Gkioxari, P. Dollar, and R. Girshick. 2017. Mask R-CNN. Page in Proceedings of the IEEE International Conference on Computer Vision (ICCV).

He, K., X. Zhang, S. Ren, and J. Sun. 2016. Deep residual learning for image recognition. Pages 770–778 in Proceedings of the IEEE conference on computer vision and pattern recognition.

Keselman, L., J.I. Woodfill, A. Grunnet-Jepsen, and A. Bhowmik. 2017. Intel(R) RealSense(TM) Stereoscopic Depth Cameras. Pages 1267–1276 in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).

Kingma, D.P., and J. Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

LeCun, Y., Y. Bengio, and G. Hinton. 2015. Deep learning. Nature 521:436–444. doi:10.1038/nature14539.

Lee, D.-H. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. Page 896 in Workshop on challenges in representation learning, ICML. Atlanta.

Momoki, Y., A. Ichinose, Y. Shigeto, U. Honda, K. Nakamura, and Y. Matsumoto. 2022. Characterization of Pulmonary Nodules in Computed Tomography Images Based on Pseudo-Labeling Using Radiology Reports. IEEE Transactions on Circuits and Systems for Video Technology 32:2582–2591. doi:10.1109/TCSVT.2021.3073021.

Oliver, A., A. Odena, C.A. Raffel, E.D. Cubuk, and I. Goodfellow. 2018. Realistic Evaluation of Deep Semi-Supervised Learning Algorithms. Page in Advances in Neural Information Processing Systems. Curran Associates, Inc.

Qiao, Y., T. Xue, H. Kong, C. Clark, S. Lomax, K. Rafique, and S. Sukkarieh. 2022. One-Shot Learning with Pseudo-Labeling for Cattle Video Segmentation in Smart Livestock Farming. Animals 12. doi:10.3390/ani12050558.

Sandler, M., A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. 2018. MobileNetV2: Inverted Residuals and Linear Bottlenecks. Page arXiv:1801.04381 in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Wang, Y., Q. Yao, J.T. Kwok, and L.M. Ni. 2020. Generalizing from a Few Examples: A Survey on Few-shot Learning. ACM Comput. Surv. 53. doi:10.1145/3386252.

Wu, Y., Y. Lin, X. Dong, Y. Yan, W. Ouyang, and Y. Yang. 2018. Exploit the Unknown Gradually: One-Shot Video-Based Person Re-identification by Stepwise Learning. Pages 5177–5186 in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition.

Xiao, J., G. Liu, K. Wang, and Y. Si. 2022. Cow identification in free-stall barns based on an improved Mask R-CNN and an SVM. Comput Electron Agric 194:106738. doi:https://doi.org/10.1016/j.compag.2022.106738.

Yao, C., X. Zhu, and K.A. Weigel. 2016. Semi-supervised learning for genomic prediction of novel traits with small reference populations: an application to residual feed intake in dairy cattle. Genetics Selection Evolution 48:84. doi:10.1186/s12711-016-0262-5.

Yao, L., Z. Hu, C. Liu, H. Liu, Y. Kuang, and Y. Gao. 2019. Cow face detection and recognition based on automatic feature extraction algorithm. Page in Proceedings of the ACM Turing Celebration Conference - China. Association for Computing Machinery, New York, NY, USA.

Yukun, S., H. Pengju, W. Yujie, C. Ziqi, L. Yang, D. Baisheng, L. Runze, and Z. Yonggen. 2019. Automatic monitoring system for individual dairy cows based on a deep learning framework that provides identification via body parts and estimation of body condition score. J Dairy Sci 102:10140–10151. doi:https://doi.org/10.3168/jds.2018-16164.

Zhang, Y., I.R. Porter, M. Wieland, and P.S. Basran. 2022. Separable Confident Transductive Learning for Dairy Cows Teat-End Condition Classification. Animals 12. doi:10.3390/ani12070886.

Zhao, K., and D. He. 2015. Recognition of individual dairy cattle based on convolutional neural networks. Nongye Gongcheng Xuebao/Transactions of the Chinese Society of Agricultural Engineering 31:181–187. doi:10.3969/j.issn.1002-6819.2015.05.026.

Zhou, Y., and X. Li. 2020. Unsupervised Self-training Algorithm Based on Deep Learning for Optical Aerial Images Change Detection arXiv:2010.07469. doi:10.48550/arXiv.2010.07469.

Zin, T.T., C.N. Phyo, P. Tin, H. Hama, and I. Kobayashi. 2018. Image technology based cow identification system using deep learning. Pages 236–247 in Proceedings of the international multiconference of engineers and computer scientists.

Zoph, B., V. Vasudevan, J. Shlens, and Q. V Le. 2018. Learning Transferable Architectures for Scalable Image Recognition. Page arXiv:1707.07012 in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

## TABLES AND FIGURES

**Table 3.1.** Capture dates and total number of images contained in each dataset split. Care was taken to ensure that images contained in the training, validation, and test sets were collected in different days, simulating a realistic scenario where a model is trained on certain dates and its accuracy is tested on future dates.

| Dataset split | Initial date | Final date | Number of images |
|---|---|---|---|
| Training | August 8th | August 9th | 2,354 |
| Validation | August 10th | August 20th | 1,161 |
| Test | September 2nd | October 7th | 1,180 |
| Unlabeled | August 21st | September 1st | 20,194 |

**Table 3.2.** Best predictive accuracy, time to train the baseline model, and the minimum and maximum training times for the first round of pseudo-labeling for each architecture. The Xception architecture provided a good trade-off between predictive and computational performance, so we decided to further investigate only this architecture in the subsequent experiments.

| Architecture | Baseline training time (minutes) | Minimum training time (minutes) | Maximum training time (minutes) | Baseline test accuracy (%) | Best test accuracy (%) |
|---|---|---|---|---|---|
| Xception | 170 | 224 | 1,350 | 77.5 | 90.2 |
| MobileNetV2 | 146 | 132 | 953 | 75.0 | 85.7 |
| NASNet Large | 200 | 331 | 2,341 | 77.0 | 91.3 |

**Table 3.3.** Training set size, test set accuracy, and percentage of images utilized (considering manually labeled and unlabeled images dedicated for training) before any pseudo-labeling and after performing four rounds of pseudo-labeling. Even starting with as few as 5% of the total images available, performing pseudo-labeling allowed for up to 94% of the images to be retrieved, labeled, and used for training. The resulting neural networks achieved a relative increase in accuracy between 20 and 40% when compared to the networks trained without pseudo-labeling. a Test accuracy after performing four rounds of pseudo-labeling.

| Dataset | Baseline training set size | Baseline test accuracy (%) | Final training set size | Final test accuracy[1] (%) | % Test accuracy increase | Initial % of images utilized | Final % of images utilized |
|---|---|---|---|---|---|---|---|
| 10% | 233 | 33.6 | 14,424 | 42.2 | 26 | 1 | 64 |
| 25% | 585 | 51.5 | 18,972 | 71.9 | 40 | 3 | 84 |
| 50% | 1,177 | 70.9 | 21,138 | 89.7 | 27 | 5 | 94 |
| 75% | 1,769 | 71.4 | 21,920 | 89.7 | 26 | 8 | 97 |
| 90% | 2,123 | 74.2 | 22,280 | 91.3 | 23 | 9 | 99 |
| full | 2,354 | 77.5 | 22,418 | 92.7 | 20 | 10 | 99 |

[1]Test accuracy after performing four rounds of pseudo-labeling

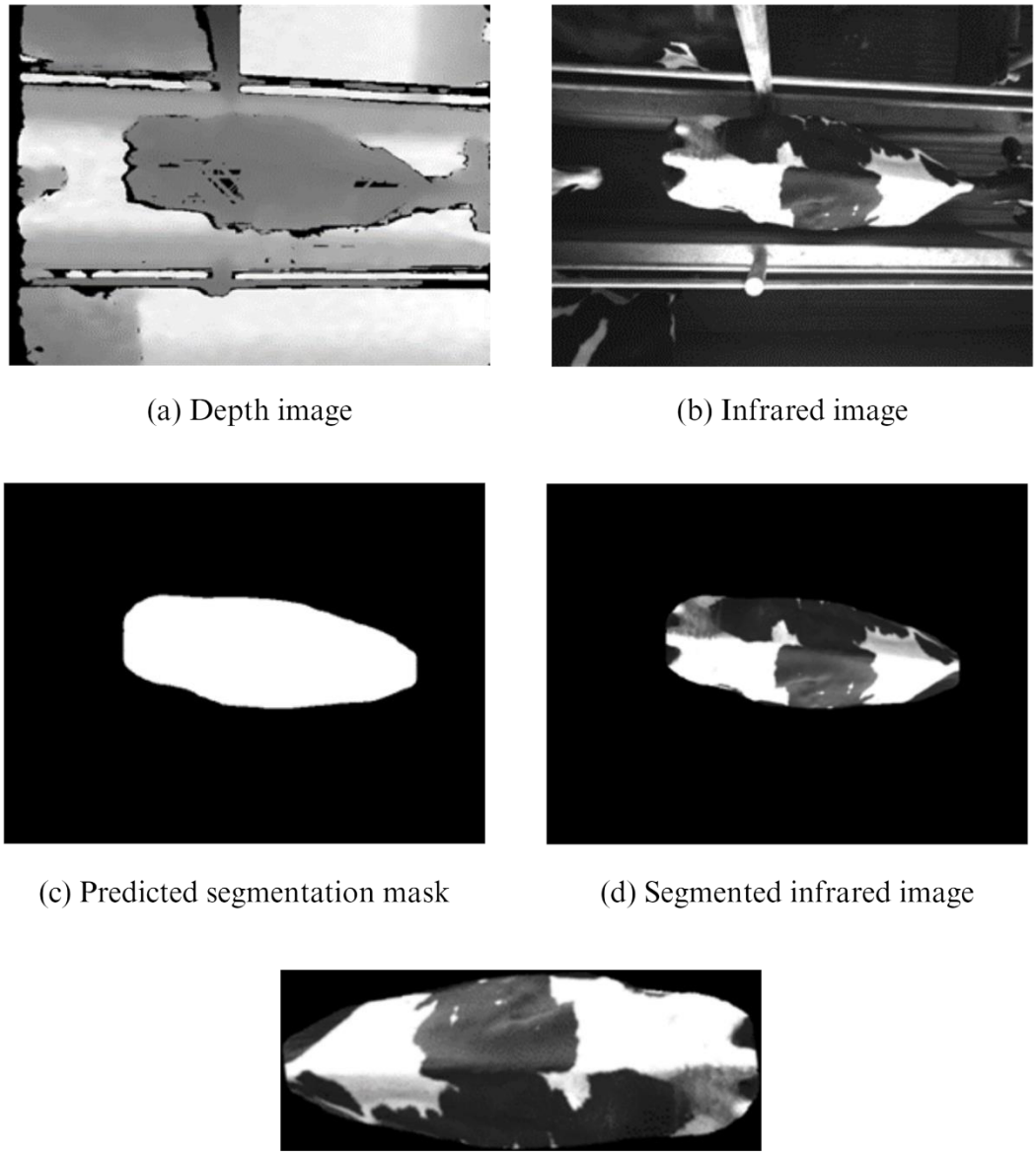



**Figure 3.1.** Example of a snapshot after each preprocessing stage: (a) shows the original captured depth image; (b) shows the original captured infrared image; (c) shows the predicted segmentation mask generated from the trained Mask R-CNN algorithm; (d) shows the segmented infrared image, after applying the predicted segmentation mask to the original infrared image; and (e) shows the resulting image from cropping and rotating (d) to only contain the area around the cow.

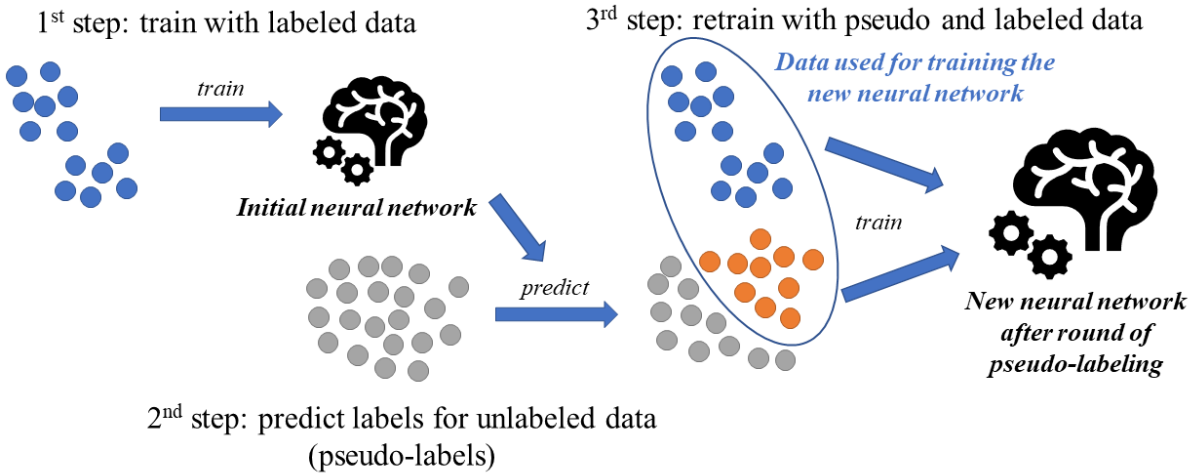



**Figure 3.2.** One round of pseudo-labeling, comprising of: training an initial neural network; running predictions on unlabeled data; and training a new neural network using both initial labeled data and unlabeled data with confident predictions, using the corresponding predicted classes as labels (pseudo-labels). Blue points correspond to labeled data, gray points correspond to unlabeled data, and orange points correspond to originally unlabeled data whose prediction confidence is greater than a given threshold. In the third step, such unlabeled images (orange points) are assigned their predicted classes as labels and are added to the training set for training a new neural network.

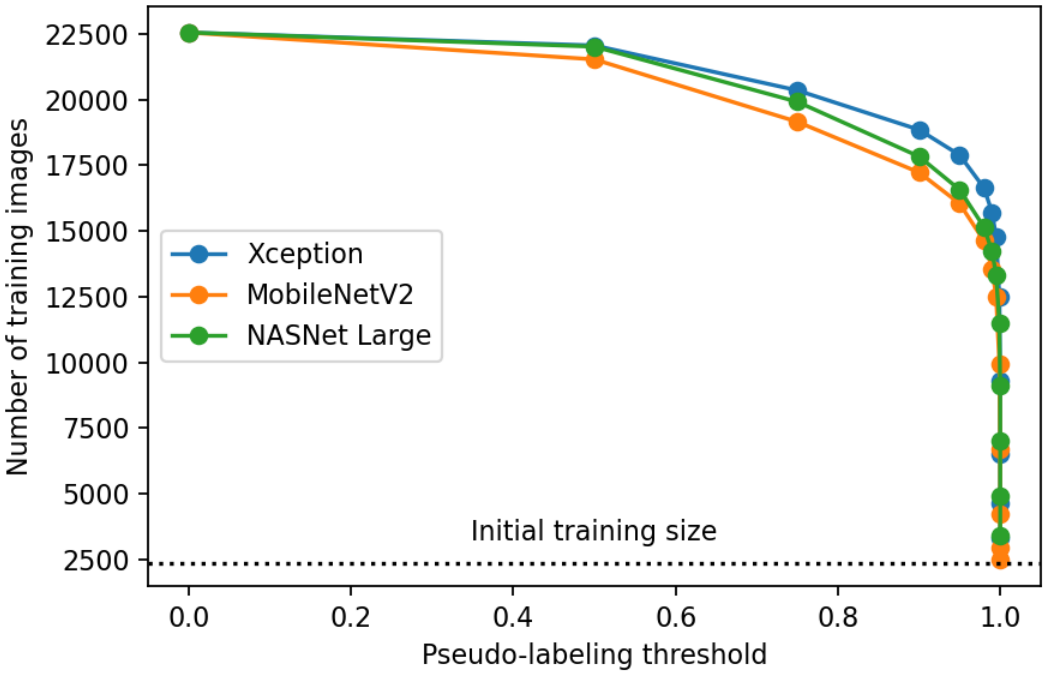



**Figure 3.3.** Number of resulting training images after filtering unlabeled data predictions using different threshold values. Threshold values were set to 0, 0.5, 0.75, 0.90, 0.95, 0.98, 0.99, 0.999, 0.9999, 0.99999, 0.999999, 0.9999999. Unlabeled images are filtered based on the prediction confidence resulting from the trained baseline model, which corresponds to the highest value in a neuron from the output layer after applying the softmax function (Eq. 3.1). Higher threshold values restrict the images used in the next training round to only those that contain pseudo-labels with higher confidence, decreasing the training set size but potentially increasing the quality of the pseudo-labels.

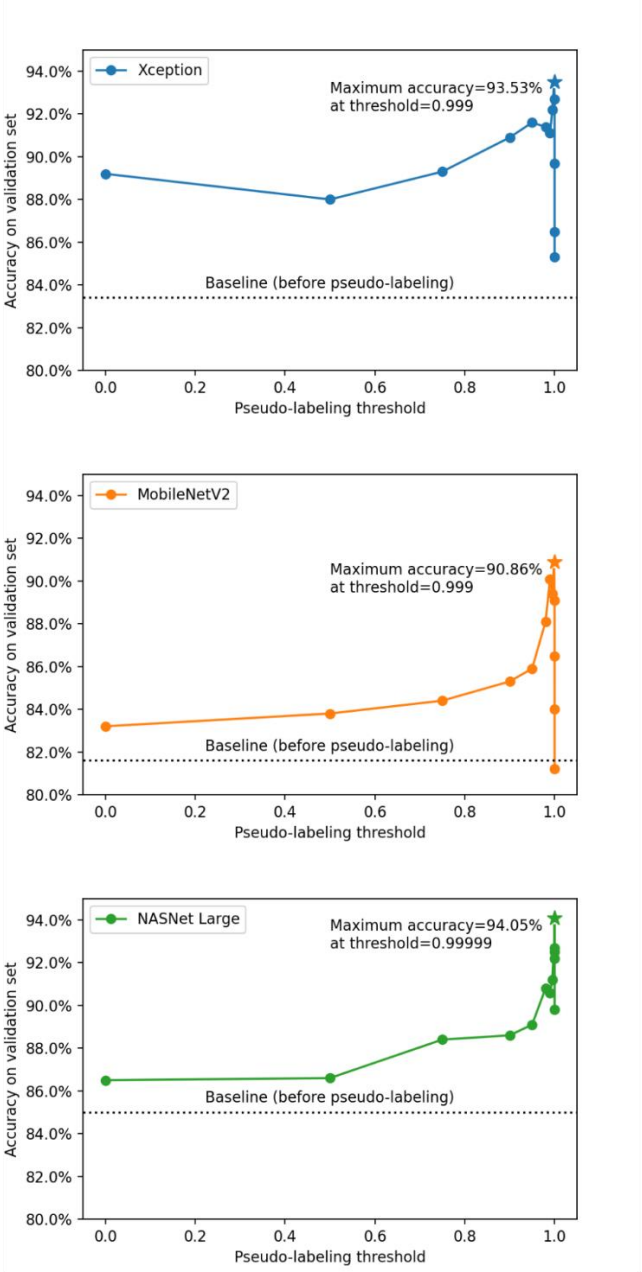



**Figure 3.4.** Validation set accuracy of the neural networks following each evaluated architecture, trained using images filtered based on different confidence threshold values. Threshold values were set to 0, 0.5, 0.75, 0.90, 0.95, 0.98, 0.99, 0.999, 0.9999, 0.99999, 0.999999, 0.9999999. The best models for each architecture, represented with stars, were trained using both manually labeled images and unlabeled images (and their corresponding predicted labels) with confidence predictions above the optimal thresholds using the corresponding baseline model. Finding the best threshold value is key to the success of applying pseudo-labeling, as lower threshold values tend to add too many noisy (and possibly wrong) pseudo-labels, and higher threshold values tend to excessively restrict the addition of unlabeled data, approaching the results achieved with the baseline model.

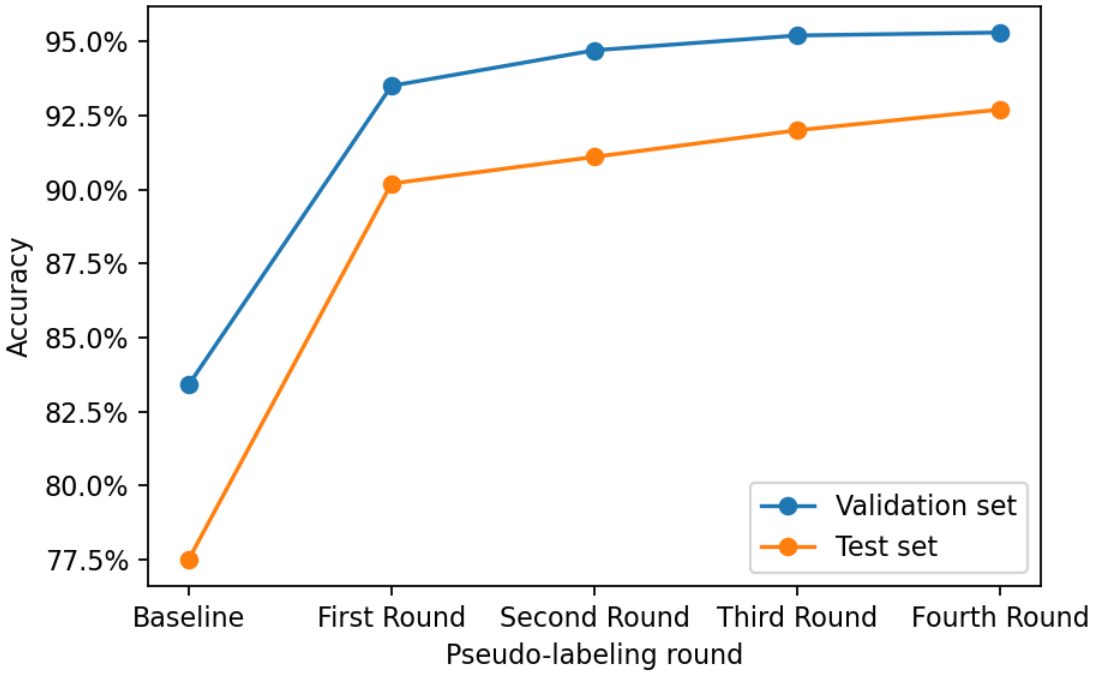
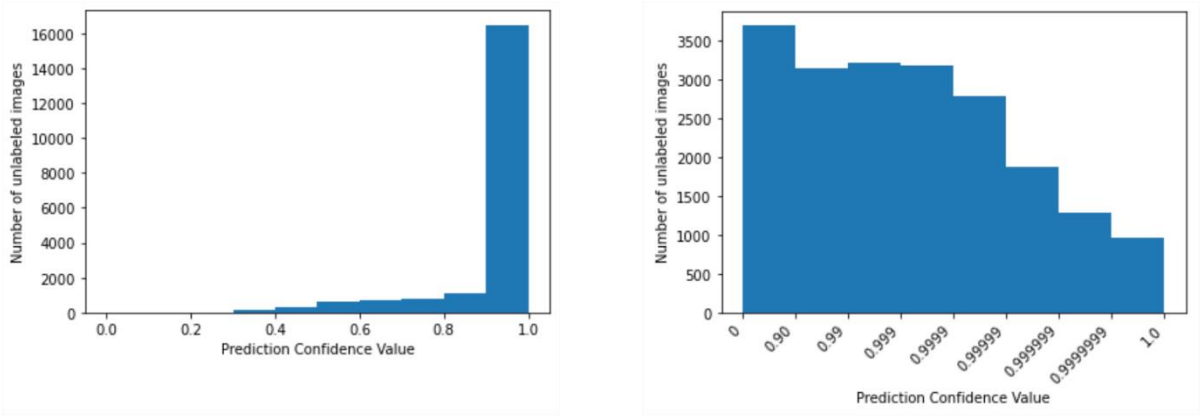



**Figure 3.5.** Accuracy on the validation and test sets of the trained networks after one, two, three, and four rounds of pseudo-labeling using the best threshold values in each round. The performance increases considerably after a single round of pseudo-labeling and remains roughly steady after the subsequent rounds.




(a) Histogram with evenly distributed bins (b) Histogram with bins between 0.9 and 1.0

**Figure 3.6.** Distribution of the confidence values predicted by the baseline fully supervised Xception model on the unlabeled dataset illustrated through a histogram containing evenly distributed bins of size 0.1 (a), and through a histogram with bins between 0.9 and 1.0 to better indicate the distribution of confidence values closer to 1 (b). Although not numerically equal, bins in (b) were set to visually have equal widths for illustration purposes.

# CHAPTER FOUR: EARLY DETECTION OF SUBCLINICAL KETOSIS IN DAIRY COWS – INTEGRATING IMAGE AND TEXT INTO A MULTIMODAL MACHINE LEARNING PIPELINE

## ABSTRACT

Computer vision systems (**CVS**) and wearable sensors can generate high-throughput animal-level phenotypes that can be used to monitor potential health problems, animal growth, and welfare. CVS can monitor minute changes in a cow's body shape, and wearable sensors can capture their behavioral patterns, which are both associated with the risk of a dairy cow developing metabolic disorders in the transition period. The objectives of this study were (1) to explore different computer vision techniques for extracting body shape features from depth images of dairy cows, and (2) to combine tabular data with features extracted from images and text into a machine learning pipeline for the early detection of subclinical ketosis (**SCK**). The proposed machine learning system combines data collected exclusively during prepartum to determine the risk of SCK during the first 15 days of lactation, defined based on the plasma beta-hydroxybutyrate (**BHB**) concentrations measured during that period. A total of 276 depth videos from 92 cows were individually collected once a week from 21 to 7 days prior to calving. From each video, 50 frames were extracted and processed using three different approaches. After removing background pixels, features were extracted from each frame using a (1) convolutional neural network (**CNN**) trained for body condition score (**BCS**) prediction, and (2) sampling depth values between automatically detected keypoints on the body surface of the cows. Features were also extracted from sequences of frames using (3) a CNN coupled with a recurrent neural network (**CNN-RNN**) for future BCS prediction. Tabular data included cow history information and prepartum feeding behavior and

activity, while unstructured text data included notes recorded in the farm management software and the structured tabular data converted to unstructured text. Features were extracted from text using large language models (**LLM**) for embedding extraction, allowing information contained in textual data to be included in the SCK prediction machine learning pipeline. The features extracted from each data source were combined to train Random Forest (**RF**) models for postpartum SCK prediction. To perform model evaluation, 20 random sets of 19 cows were sampled with replacement, determining 20 independent test sets. For each test set, data from the remaining cows were used to train and perform hyperparameter tuning on the RF models using 5-fold cross-validation. The best RF models trained using only image features achieved $F_1$ scores of 0.413 (± 0.182), 0.455 (± 0.153), and 0.493 (± 0.110) when extracting features via the BCS CNN, anatomical keypoints, and future BCS CNN-RNN approaches, respectively. The best RF models integrating image features and tabular data achieved an average $F_1$ score of 0.706 (± 0.125), which was superior to the models trained using only tabular data ($F_1$ score = 0.655 ± 0.094). When combining tabular and text data, the best RF models achieved an average $F_1$ score of 0.681 (± 0.209), which was also greater than the one achieved using only tabular data. These results indicate that integrating image, text, and tabular data representing prepartum body shape, feeding behavior, cow activity, and cow history information via the proposed machine learning pipeline can be a powerful tool for the early detection of SCK in dairy cows in an automated manner. Additionally, leveraging modern deep learning techniques for extracting features from high-dimensional unstructured data such as images and text proved to be beneficial for improving the performance of the trained models. The proposed system could allow the implementation of preventive practices in dairy farms, reducing costs associated with subclinical ketosis, and improving animal health and welfare.

## INTRODUCTION

Transition dairy cow metabolism, management, and prevention of peripartum diseases have been the primary focus of dairy cattle research over the past two decades (Drackley, 1999; LeBlanc, 2010; Cardoso et al., 2020). During the transition period, dairy cows usually experience negative energy balance (**NEB**) to support the high energy demands of lactation. The severity of NEB can increase the risk of a variety of peripartum disorders, such as retained placenta (Cameron et al., 1998), metritis (Hammon et al., 2006), endometritis (Dubuc et al., 2010), displaced abomasum (LeBlanc et al., 2005), ketosis (Green et al., 1999), and hypocalcemia (Horst et al., 1994). Among the peripartum disorders associated with severe NEB, ketosis is one of the most prevalent, and it causes large economic losses on dairy farms due to costs of treatment, reduced productive and reproductive performance, and increased culling rates (Steeneveld et al., 2020). Moreover, the greater incidence of health problems negatively impacts animal welfare, longevity, and public perception of the dairy industry.

According to (Grummer, 1993), an important indicator of NEB is an elevation in plasma non-esterified fatty acid (**NEFA**) concentrations. However, frequently assessing accurate plasma NEFA values can be challenging, as it requires blood tests or milk fatty acids analysis (Jorjong et al., 2014; Dórea et al., 2017; Menezes et al., 2024), which can be costly or labor-intensive to perform. Similarly, current methods for detecting ketosis events in large farm operations may not be as reliable as laboratory analyses (Wilson and Goodell, 2013; Lei and Simões, 2021) and are also labor intensive, as they require collecting blood or milk samples from the cows to be tested (Enjalbert et al., 2001; de Roos et al., 2007). Additionally, to the best of our knowledge, there are no proposed automated methods in the literature for assessing, prior to calving, the risk of a cow developing subclinical ketosis during early lactation.

Body condition score (**BCS**) can be used as a tool to assess the impact of NEB in early lactating cows. A high BCS in prepartum cows is associated with greater risks of health disorders and poorer reproductive performance, primarily due to elevated body fat mobilization (Buckley et al., 2003; Overton and Waldron, 2004; Roche et al., 2009, 2015). However, BCS is highly subjective and prone to inconsistencies across different evaluators, and even within the same evaluator. To address this issue, computer vision systems (**CVS**) have been developed to assess BCS in a more systematic way (Qiao et al., 2021). Nevertheless, these computer vision systems are trained using human-generated assessments, so they retain some of the subjectivity from the evaluators. Additionally, the standard quarter-point divisions used for evaluating BCS do not enable the detection of subtle changes in body shape, or the distinction between different body shapes that fit in the same quarter-point category (for example, two animals might have different fat distributions around the hooks and the pins, and still be given the same BCS). A way to overcome these limitations is to extract, from images captured from the animals, geometric features that characterize body shape change, since such features are objective values that do not rely on human evaluation.

Previous studies demonstrated that depth images can generate precise biometric measurements related to body shape, such as volume, torso area, length, height, and width, in pigs (Fernandes et al., 2019, 2020) and cattle (Cominotte et al., 2020), as well as features specifically related to body fat stores in dairy cattle (Song et al., 2019; Liu et al., 2020; Zin et al., 2020). Furthermore, convolutional neural networks (**CNN**) have been used to automatically extract nonlinear and invariant features from depth images in the context of livestock farming (Borges Oliveira et al., 2021; Caffarini et al., 2022). The features extracted using such method are often difficult to interpret directly, but they represent important characteristics of the objects contained

in the image and can be used in combination with other machine learning techniques to perform classification tasks (Andrew et al., 2017; Qiao et al., 2019). Thus, CNNs trained to predict BCS, such as the one proposed by Yukun et al. (2019), could be used to extract features related to body shape.

Feeding and activity behavior during peripartum can be early indicators of subclinical ketosis in dairy cows (González et al., 2008; Goldhawk et al., 2009; Itle et al., 2015). Such behavioral data can be automatically collected using electronic roughage intake control bins and activity monitoring ear tags. Additionally, farm employees usually record textual notes in their farm management software including pen moves, vaccine and medication administration, pregnancy checks, insemination procedures, health events, and other management information. Recently, large language models (**LLM**) have achieved great success in extracting quantitative information from text in the form of contextual embeddings for knowledge retrieval, anomaly detection, text classification, and text clustering (Min et al., 2023). Such embeddings extracted from cow-specific text notes recorded in farm management software could potentially contain important information that helps predict the risk of diseases in the herd.

The objectives of this study were (1) to explore different computer vision techniques for extracting body shape features from depth images of dairy cows, and (2) to combine tabular data with features extracted from images and text into a machine learning pipeline for the early detection of subclinical ketosis (**SCK**). The proposed machine learning system combines data collected exclusively during prepartum to determine the risk of SCK during the first 15 days of lactation, defined based on the plasma beta-hydroxybutyrate (**BHB**) concentrations measured during that period.

## MATERIAL AND METHODS

The proposed machine learning pipeline consisted of (1) using deep learning and image processing techniques to extract features related to body shape from depth images collected from dairy cows during prepartum; (2) calculating descriptive variables from prepartum feeding behavior, cow activity, and cow history data; (3) extracting features from textual data using LLMs; and (4) integrating all the extracted features into machine learning models that predict, using exclusively prepartum data, the cows with a high risk of developing subclinical ketosis during the first 15 days of lactation. In this last step, we evaluated the impact of using different feature extraction approaches for image and textual data on the predictive performance of the models. Figure 4.1 illustrates these steps.

While structured tabular data can generally be used directly to train machine learning models for phenotype prediction, unstructured data such as images and text require a feature extraction step to convert them into lower-dimensional structured features first. Exploring different approaches for extracting features from image and textual data was a crucial step in the proposed machine learning pipeline, as it enabled the inclusion of important information originating from this high-dimensional unstructured data that would otherwise not be possible to be used in predictive modeling.

Image Feature Extraction

Three different approaches were evaluated for extracting features from prepartum depth images: (1) extracting the output of the second-to-last layer of a CNN trained for BCS prediction; (2) sampling depth values between automatically detected keypoints on the body surface of the cows; and (3) extracting the output of the second-to-last layer of a convolutional and recurrent neural network (**CNN-RNN**) trained for future BCS prediction using sequences of depth frames of the same cow taken on consecutive weeks.

Prepartum Depth Images Dataset

Videos from 115 multiparous Holstein cows, housed at the Emmons Blaine Dairy Cattle Research Center (Arlington, WI), were manually collected weekly from 21 to 7 days before the expected calving date, and from 7 to 56 days after calving. Utilizing an Intel RealSense D435 depth-sensing camera (Keselman et al., 2017) positioned about 5 meters above the scale during individual weighing sessions, the videos displayed a top-down perspective of the back of the animals, capturing a 3-dimensional (**3D**) representation of their body surface. These recordings were conducted between December 22$^{nd}$, 2020, and June 4$^{th}$, 2021. Not all cows had all 11 videos captured, because they either calved more than a week before their expected calving date, or left the trial earlier, which resulted in a total of 1,164 videos. The videos were recorded at a resolution of 848 × 480 pixels and 60 frames per second, and they ranged between 10 and 20 seconds long. From each video, 50 random depth frames were extracted, each at a resolution of 848 × 480 pixels, with each pixel encoded as a 16-bit unsigned integer representing the distance in millimeters between the camera and the object in that pixel.

CNN Models for BCS Prediction

The first approach for extracting features from the depth frames was to extract features from the second-to-last layer of a CNN trained for BCS prediction. The hypothesis behind this approach was that a CNN that can predict BCS would indirectly learn to extract relevant features related to the body shape of the cows. These features could be more informational than just the BCS value itself for predicting phenotypes related to body shape changes and body fat mobilization, such as the risk of developing subclinical ketosis.

CNNs were trained for BCS prediction using depth frames extracted from the videos collected weekly during individual weighing. In the same days as the videos were collected, three

trained independent evaluators assessed the BCS of each cow using a 5-point scale (Wildman et al., 1982). The BCS in quarter-point increments that was closest to the average among the three evaluators was determined for each video, representing an individual cow and a date. The assessed BCS values were 2.00 (n = 1), 2.25 (n = 18), 2.50 (n = 42), 2.75 (n = 123), 3.00 (n = 222), 3.25 (n = 247), 3.50 (n = 218), 3.75 (n = 139), 4.00 (n = 84), 4.25 (n = 43), and 4.50 (n = 27).

Each depth frame extracted from the videos was processed using the steps illustrated in Figure 4.2 and described as follows. The 10 preceding and 10 subsequent depth frames were collected from the original video – these frames were utilized to reduce individual pixel noise and perform a temporally-based depth value denoising. For each frame, including the central and the 20 adjacent ones, pixels containing a value of 0, which usually represent an error in the computed depth value, were filled with the value in the closest non-zero pixel. Masks containing the cow body, excluding the neck and the head, were extracted from each depth frame using a semantic segmentation neural network based on the U-net architecture (Ronneberger et al., 2015). This cow body segmentation neural network was trained using 252 depth frames from 84 animals and tested on 80 depth frames from other 27 animals, achieving an average intersection over union (**IoU**) of 0.960 and an average Dice similarity coefficient of 0.979 on the testing set. The predicted masks were applied to each corresponding depth frame, setting every pixel outside of the masks to 0. Each frame was rotated so that the major axis of the ellipse that had the same second-moments of the mask was parallel to the x-axis, and cropped around the bounding box containing all mask pixels with a 5-pixel padding on each side. The mean pixel values of the 21 rotated and cropped frames (one central and 20 adjacent) were set as the final denoised depth frame, and a value of 0 was assigned to any pixel that contained 0 in at least one of the 21 frames. This final denoised depth frame was normalized to values between 0.1 and 1.0 using the minimum and maximum pixel

values excluding 0 and converted to an 8-bit image by multiplying the resulting pixel values by 255.

The resulting 58,200 denoised depth frames were used to train and validate CNNs for BCS prediction following three different paradigms: a regular multi-class image classification neural network with a softmax output layer containing 11 neurons, one for each BCS value; and neural networks for rank-consistent ordinal regression following the Consistent Rank Logits (**CORAL**) architecture and loss function (Cao et al., 2020), and the Conditional Ordinal Regression for Neural Networks (**CORN**) training procedure and loss function (Shi et al., 2023). Rank-consistent ordinal regression methods consider the relative ordering between labels, as it exists in body condition scoring, while producing consistently ranked predictions.

All networks contained a ResNet-50 backbone (He et al., 2016) and were trained via transfer learning starting from a ResNet-50 network pretrained on the ImageNet dataset (Deng et al., 2009), with the final classification layers modified according to each training paradigm. The transfer learning training procedure was performed following a two-stage approach: first, only the final classification layers of the networks were trained for 30 epochs, keeping the weights from all layers of the convolutional ResNet-50 backbone frozen; then, all weights were unfrozen, and the networks were trained for 60 epochs with a smaller learning rate. The batch size was set to 8 in both stages, and the initial learning rates were $10^{-3}$ and $10^{-4}$ for each stage respectively. The weights of the networks were optimized for minimizing the corresponding loss function using Adam (Kingma and Ba, 2014) with running average coefficients of 0.9 and 0.999, and a scheduler was set to decrease the learning rate by a factor of 10 every 6 epochs. During training, the images were resized to 224 × 224 and randomly flipped horizontally and vertically with 50% probability. An independent test set was defined containing 11,550 depth frames of 21 of the 115 cows, and the

remaining 46,650 frames were used to train the neural networks. The performance of the models was assessed by evaluating the accuracy for predicting the exact BCS quarter-point values, as well as the accuracies considering error tolerances of 0.25, 0.50, 0.75, and 1.0.

For each model, 2,048 features were extracted from the output of the ResNet-50 backbone, which were later used for subclinical ketosis prediction. Features from one random frame of each prepartum video were concatenated, resulting in 6,144-dimensional feature vectors for each cow. Since 50 frames were extracted from each video, there were $50^3$ possible variations of feature vectors for each cow, but only up to 50 of those variations were used, per cow, for subclinical ketosis prediction.

Body Surface Between Anatomical Keypoints

Another approach for image feature extraction evaluated in this study was to sample the body surface of the cows between anatomical keypoints. This approach more directly extracts body shape information from the depth frames than the BCS CNN method, as it consists of sampling the pixel depth values between predetermined keypoints over the cow body instead of indirectly extracting features from a network trained for the related task of BCS prediction.

Eight keypoints were determined on the bodies of the cows, located at specific anatomical landmarks visible from the top-down perspective captured in the depth videos. The keypoints included the (1) left and (2) right hooks, (3) left and (4) right pin bones, (5) tailhead, (6) sacral vertebrae, (7) lumbar vertebrae, and (8) cervical vertebrae, as shown in Figure 4.3. A YOLOv8 model (Jocher et al., 2023) was trained to automatically detect those keypoints in the collected depth images. This keypoint detection model was trained for 100 epochs using 29,626 depth images of 77 cows and tested on 7,088 depth images of a different set of 17 cows, which also belonged to the set of cows used for testing the BCS prediction models. The model predicted

keypoints with an average error of 7.37 pixels on the testing set, calculated as the Euclidean distance between each predicted keypoint and its corresponding ground truth label. This corresponds to an average error of 0.022 when calculated using normalized image dimensions.

With the eight keypoints detected for each depth frame, 1-dimensional (**1D**) vectors were calculated by linearly sampling depth values between keypoint pairs, as illustrated in Figure 4.4. The pairs of keypoints used for extracting such vectors were (1 and 8), (2 and 8), (1 and 7), (2 and 7), (1 and 6), (2 and 6), (2 and 4), (1 and 3), (4 and 5), (3 and 5), and (3 and 4). The regions between such keypoints represent areas that are frequently looked at when assessing the body condition score of dairy cows, as these areas usually go through a considerable visual change as the body fat stores of the cows vary, thus changing the depth values sampled across them as well. These multiple depth vectors calculated from each pair of keypoints were concatenated to form a set of features extracted from each depth frame. Four different sampling resolutions (*sampling_res*) were evaluated: 20, 50, 100, and 200, which represent the number of depth values sampled between each pair of keypoints. This process generated (*sampling_res×11)* features per image, with 11 being the number of keypoints pairs. Additionally, normalized versions of the features for each sampling resolution were calculated by subtracting the depth value at each sample point by the corresponding depth value of that point projected to the 3-dimensional line connecting the two keypoints in the 3-dimensional space consisting of the x and y coordinates of the keypoints, and the depth values at their location. In other words, the normalized features are the depth values (original feature values) subtracted by the values projected in the line connecting the two keypoints in the depth dimension, represented by the dashed lines in Figure 4.4. Finally, an additional set of features was calculated by extracting the area between the 1D vectors and the lines connecting the two keypoints for each pair of keypoints. This set of features was calculated with sampling

resolution equal to 200, as it was the highest resolution evaluated, and generated 11 features per image, with 1 feature per pair of keypoints.

The features extracted from one random frame of each prepartum video were concatenated into 660- to 6,600-dimensional feature vectors for each cow. Since 50 frames were extracted from each video, there were $50^3$ possible variations of feature vectors for each cow, but only up to 50 of those variations were used, per cow, for subclinical ketosis prediction.

CNN-RNN Models for Next-Week BCS Prediction

The last approach explored for extracting features from depth images was to extract the output of the second-to-last layer of a CNN-RNN model trained for predicting the BCS of the cow during the following week using sequences of depth frames of that same cow taken on previous consecutive weeks. The hypothesis to be evaluated was that this CNN-RNN model would be able to capture not only body shape information from each depth frame independently via its CNN portion, but also extract relationships between depth frames taken over time, which could be indicative of body shape changes and, consequently, body fat mobilization. The model was trained for future BCS prediction to force it to potentially use information from all the images of the sequence, instead of simply using only the image corresponding to the target BCS if the target was defined as the BCS at the time that the last image of the sequence was taken, for example.

CNN-RNNs were trained to predict the BCS of a cow during the following week using sequences of depth frames taken on previous consecutive weeks. The target BCS values were distributed into 2.00 (n = 1), 2.25 (n = 18), 2.50 (n = 42), 2.75 (n = 123), 3.00 (n = 217), 3.25 (n = 231), 3.50 (n = 194), 3.75 (n = 115), 4.00 (n = 60), 4.25 (n = 28), and 4.50 (n = 20). Each sequence contained from 1 to 10 depth frames of the same cow taken on consecutive weeks.

The extracted depth frames from all except the last video of each cow was preprocessed using the same procedure described in the *CNN Models for BCS Prediction* subsection and illustrated in Figure 4.2, which resulted in 52,450 rotated, cropped, and denoised 8-bit images containing the segmented body surface of a cow originated from 1,049 videos. The last video of each cow was excluded from this analysis because there was no next-week BCS annotation for those. These images were used to construct sequences of length between 1 and 10 during training. During each training step, sequences of *n* images were randomly selected, where *n* was the length of the sequence, and each image belonged to a different video of the same cow taken on consecutive weeks. The image sampling for constructing the sequences was determined randomly because, for each sequence of *n* videos of the same cow, a total of $50^n$ different frame combinations could be defined, which makes it unfeasible to go through every single possible frame combination during training.

CNN-RNNs models following different architectures and training paradigms were evaluated for predicting the BCS of that cow on the subsequent week. The same three training paradigms as used for single-image BCS prediction (*CNN Models for BCS Prediction*) were explored: regular softmax output layer, CORAL, and CORN. The CNN backbone for extracting features from each individual image followed a ResNet-50 architecture (He et al., 2016) pretrained on ImageNet (Deng et al., 2009) or initialized with the weights from the best single-image BCS prediction model. A similar transfer learning procedure as described in the *CNN Models for BCS Prediction* subsection was implemented, but with the first stage optimizing the weights of both the output layer and the Recurrent Neural Network (**RNN**) while keeping the weights of the CNN frozen. For training the CNN-RNN models initialized with the best BCS CNN weights, we also evaluated skipping the second stage of transfer learning, keeping the CNN backbone weights

frozen during the whole training procedure. The RNN portion of the model consisted of 1 or 2 Long Short-Term Memory (**LSTM**) layers (Hochreiter and Schmidhuber, 1997) with dimensions 64, 128, 256, 512, 1,024, or 2,048. Finally, we explored allowing different minimum and maximum sequence lengths, and including or not postpartum images during training.

The batch size was set to 8 sequences in both stages of transfer learning, and the initial learning rates were set to $10^{-3}$ and $10^{-4}$ for each stage respectively. The weights of the networks were optimized for minimizing the corresponding loss function using Adam (Kingma and Ba, 2014) with running average coefficients of 0.9 and 0.999, and a scheduler was set to decrease the learning rate by a factor of 10 every 6 epochs. During training, the images were resized to 224 × 224 and randomly flipped horizontally and vertically with 50% probability, and sequences of length between the minimum and maximum allowed were determined by randomly selecting frames from the different corresponding videos. An independent test set was defined containing every sequence from the same 21 cows used for testing the single-image BCS prediction CNN and the keypoint detector, which resulted in a total of 210 video sequences that were randomly sampled as frame sequences during model evaluation. The 839 video sequences from the remaining 94 cows were used to randomly sample frame sequences during training. Similarly to the single-image BCS CNN, the performance of the models was assessed by evaluating the accuracy for predicting the exact next-week BCS quarter-point values, as well as the accuracies considering error tolerances of 0.25, 0.50, 0.75, and 1.0.

For each model, two feature extraction methods were explored: extracting features from the full input sequence by retrieving the last (time-wise) hidden state of the last (depth-wise) LSTM layer; or extracting features from each image of the input sequence individually by retrieving all the hidden states of the last (depth-wise) LSTM layer. Since RNNs process each element of a

sequence taking into consideration information remaining from the previous elements, features extracted from a sequence of images might take advantage of the relationships between images of the same cow taken in consecutive weeks, potentially leading to features that are better subclinical ketosis predictors than using the single-image CNN for BCS prediction approach. When extracting features from the full input sequence, each sequence of three depth frames (one from each prepartum video) resulted in a 64- to 2,048-dimensional feature vector. When extracting features from each image of the input sequence individually, features from each image were concatenated, resulting in 192- to 6,144-dimensional feature vectors for each cow. Since 50 frames were extracted from each video, there were $50^3$ possible combinations of prepartum frames to form a sequence, and thus $50^3$ possible variations of feature vectors for each cow, but only up to 50 of those variations were used, per cow, for subclinical ketosis prediction.

Tabular Data – Behavior and Cow History

Of the 115 cows, 23 did not have at least three videos recorded prepartum and were excluded from further analyses. For the remaining 92 cows, the following data was retrieved from the management software at the farm: parity, days in milk of the previous lactation, previous dry period length, number of past ketosis events, and weekly BCS and body weight in the last three weeks before calving. Electronic roughage intake control bins (Hokofarm Group; Marknesse, the Netherlands) measured the weight and duration of all meals from these animals between 21 days before the expected calving date and the actual calving date. The daily averages of dry matter intake (**DMI**), feeding time, average meal duration, and number of meals were calculated for both 7 days and 2 days prior to the calving date for each cow. Additional behavioral data were collected via SMARTBOW (Zoetis; Kalamazoo, MI) ear tags fitted to each cow, including lying time, rumination time, and time spent inactive and highly active, and daily averages were calculated for

the last 7 days before the calving date. This resulted in 25 feeding behavior, cow activity, and cow history variables per cow, as shown in Table 4.3, which constituted the tabular data utilized in this study.

Text Feature Extraction

Feeding behavior data, cow activity, and cow history variables were converted to unstructured text by using 5 different templates, illustrated in Table 4.2. The variable values were inserted into the templates in their corresponding position, resulting in 5 different texts for each cow. Text embeddings were then extracted from each generated text using the pretrained *text-embedding-ada-002* model (OpenAI; San Francisco, CA), resulting in a 1,536-dimensional feature vector for each text. Text embeddings are numerical representations of text, ideally capturing relationships between different concepts included in that text and facilitating quantitative analyses on different pieces of textual data. The *text-embedding-ada-002* model was trained on a large text corpus and was specifically optimized to extract text embeddings for text search, sentence similarity, code search, and text classification tasks, and is part of the suite of embedding models made available via Open AI's application programming interface (**API**). The purpose behind creating these texts and extracting text embeddings from them was to compare the effectiveness of incorporating behavioral and historical information in both tabular and textual formats (via text embeddings) when training machine learning models for SCK prediction.

Notes recorded during the previous lactation and dry period of each cow were collected from the farm management software, including pen moves, pregnancy checks, insemination procedures, health events, and others. These notes were exported from the management software into files in comma-separated values (**CSV**) format, which were then converted to more human-readable text using OpenAI's chat completion API. We used the *GPT-4* model, with the following

system message that provided some context to the model: *“"DIM" means the number of days in lactation that the cow had when that event happened. "PEN" is the pen number where that event occurred.”*, and the following user prompt: *“Give me a chronological report of events that happened to the cow described in this CSV: "{CSV content}"”*, with the temperature parameter set to 0.5. The temperature dictates the degree of randomness in the model’s outputs, controlling the balance between consistency (lower temperature) and diversity (higher temperature) in the text produced by the model. In addition, the calving date was appended to the end of the generated text, to provide temporal information about the start of the next lactation, for which we wanted to predict the risk of subclinical ketosis. An example of a CSV file containing the cow notes, and the corresponding converted text, is shown in Figure 4.7.

Text embeddings were extracted from each generated notes text using OpenAI’s *text-embedding-ada-002*, resulting in 1,536 features per text. Additionally, for each cow, the generated notes text was appended to each of the texts generated from the tabular data, and text embeddings were extracted using the *text-embedding-ada-002* model. This resulted in three different textual 1,536-dimensional feature sets per cow: embeddings from text converted from tabular data, generated using the templates illustrated in Table 4.2 (*template text*; 5 feature vectors per cow); embeddings from text generated from CSV files containing notes taken during the previous lactation and dry period (*notes text*; 1 feature vector per cow); and embeddings from text combining both *template text* and *notes text* (*combined text*; 5 feature vectors per cow).

Subclinical Ketosis Prediction

One of the objectives of this study was to train and validate models for predicting the risk of subclinical ketosis prediction during early lactation using exclusively prepartum data. Subclinical ketosis was defined based on blood samples collected during the first 14 days of

lactation. Blood samples from 92 cows were collected at 3, 5, 7, 11, and 14 days after the calving date. Concentrations of plasma beta-hydroxybutyrate (**BHB**) were quantified using the Catachem ChemWell-T analyzer (Catachem; Oxford, CT) as previously described (Holdorf et al., 2023). Cows with maximum measured BHB concentration among those five samples above 1.0 mmol/L were initially classified as having a postpartum subclinical ketosis event. BHB thresholds of 1.1 and 1.2 mmol/L were also evaluated, but most experiments were performed using a threshold of 1.0 mmol/L because that resulted in the most balanced dataset (of the 92 cows, 37, 28, and 21 had a maximum measured BHB value above 1.0, 1.1, and 1.2 mmol/L, respectively).

The prepartum data utilized included tabular data, features extracted from depth images, and features extracted from text constructed using the tabular data and notes recorded in the farm management software. The procedure for training and evaluating models for the early detection of subclinical ketosis was divided into seven tasks: (1) train models using only features extracted from depth images, and compare the results obtained using the different image feature extraction approaches with the ones obtained by a baseline model trained using only BCS measurements as predictors; (2) combine the best image features with the tabular data, and compare the results with baseline models containing only the tabular data; (3) train models using different number of data points per cow by using different variations of the corresponding feature vectors, and evaluate how this affected the results; (4) explore different BHB thresholds for defining subclinical ketosis, and compare training regressors for BHB prediction with training direct classifiers; (5) compare the performance of models trained using tabular data with those trained using the embeddings extracted from the text generated using the same variables (*template text*); (6) explore including information from textual notes retrieved from the management software (*notes text*) by concatenating their text embeddings with the tabular data directly, or by combining the textual

notes with the text generated from tabular data (*combined text)* and extracting text embeddings from those combined texts; and (7) combine the best features extracted from textual data and depth images with tabular data to train a final SCK prediction model.

For tasks containing tabular data, we assessed how removing dry matter intake from the analyses impacted the results, as we consider dry matter intake to be a difficult variable to measure in most commercial settings, without intake control bins available. Additionally, we evaluated performing principal component analysis (**PCA**) on the depth image and text features before training the models to match the same number of components as the tabular data (25, or 23 when excluding dry matter intake measurements).

For feature sets that contained more than one variation per cow, as shown in Table 4.3, the mean value of each feature was calculated when using one data point per cow, or random feature vectors were sampled when using more than one data point per cow. Feature sets containing $50^3$ possible variations (depth images) had 50 variations randomly sampled to calculate the mean feature vector when considering only one data point per cow. When including multiple data points per cow, the final prediction for a given cow was achieved by selecting the most predicted binary class.

For all seven tasks, Random Forest (**RF**) models (either classifiers or regressors, depending on the task) were trained using 73 cows, leaving the other 19 as testing cows. A 5-fold cross-validation procedure was performed within these 73 cows to select the best set of hyperparameters using randomized search, uniformly sampling 100 times among the values shown in Table 4.4. The set of random hyperparameters that achieved the highest cross-validation $F_1$ score (for classification), or lowest cross-validation root mean squared error (for regression) was selected to train a final model using all 73 cows. This procedure was repeated 20 times, using different training

and testing cows at each iteration, and the mean and standard deviation of the accuracy, $F_1$ score, precision, recall, and specificity were calculated for each model across the 20 iterations.

## RESULTS AND DISCUSSION

One of the goals of this study consisted of exploring different ways to extract features from depth images and textual data for use in a machine learning pipeline that predicts postpartum SCK in dairy cows using prepartum data. Before exploring the performance of the SCK prediction models, we discuss the results of the computer vision algorithms for BCS prediction, used in this study to extract features from depth images. Then, we go through the results of each of the seven tasks described in the *Subclinical Ketosis Prediction* section.

### BCS Prediction

#### CNN Models for BCS Prediction

Two of the three feature extraction approaches explored in this study for depth images relied on deep learning models for BCS prediction. The first approach consisted of extracting features from the second-to-last layer of a CNN trained for BCS prediction using segmented, rotated, cropped, and denoised depth images containing a single cow. In this approach (*CNN Models for BCS Prediction*), we explored three different training paradigms for training the BCS prediction model: (1) regular multi-class image classification using a softmax output layer, and rank-consistent ordinal regression using (2) CORAL (Cao et al., 2020) and (3) CORN (Shi et al., 2023). Using a regular softmax output layer, the model achieved accuracies of 28.4%, 66.9%, 86.6%, 96.0%, and 98.4% considering error tolerances of 0, 0.25, 0.50, 0.75, and 1.00 points, respectively. Using rank-consistent ordinal regression, the CORAL model achieved accuracies of 33.4%, 78.9%, 94.2%, 98.9%, and 99.9%, and the CORN model achieved accuracies of 31.5%, 73.3%, 90.4%, 97.7%, and 99.3% considering the same error tolerances, as shown in Table 4.5.

These results show that implementing rank-consistent ordinal regression methods for BCS prediction provides great benefits, achieving higher accuracies than regular classification using softmax, and similar accuracies to those previously reported in the literature considering a similar level of automation (Qiao et al., 2021). Achieving high accuracies at exact matches between prediction and ground truth tends to be difficult as BCS is a partially subjective measurement. Because of that, it is useful to evaluate the performance of the models when tolerating minor deviations of 0.25 to 1.0 points. Using CORAL, the high accuracies of 78.9% and 94.2% at predicting a BCS within 0.25 and 0.50 points of the ground truth label show that the trained model was able to recognize the rough body conditioning of the cows. Although the performance of direct BCS prediction might not be a guarantee that the model is a good feature extractor for downstream tasks, it is at least indicative that it might have learned to extract important information related to the body shape of the cows using depth images. The performance of the extracted features as predictors for the early detection of SCK is reported later in this section (*Models for the Early Detection of Subclinical Ketosis*).

CNN-RNN Models for Next-Week BCS Prediction

The other approach for extracting features from depth images using BCS prediction models consisted of training CNN-RNN models for next-week BCS prediction (*CNN-RNN Models for Next-Week BCS Prediction*). Using sequences of depth images extracted from videos taken in consecutive weeks, CNN-RNN models that learn spatial and temporal features were trained to predict the BCS of a cow on the week following the last frame of the sequence. Table 4.1 shows the different hyperparameters that were evaluated for constructing the CNN-RNN models, with the different comparative analysis performed as follows: (1) training paradigm, initial CNN weights, and inclusion of the second stage of transfer learning; (2) number and dimension of LSTM

layers; and (3) minimum and maximum sequence length and inclusion of postpartum images. In order to determine the best model within each comparative analysis, we evaluated the accuracies considering 0.25 error tolerance. This provided a balanced evaluation framework that was not too strict or too lenient for considering correct predictions.

The same three training paradigms used for single-image BCS prediction were explored for the next-week BCS prediction CNN-RNN models: regular softmax output, CORAL, and CORN. The CNN portion of the models were initialized either with the weights of a CNN pretrained on the general ImageNet dataset, or with the weights of the CORAL model for single-image BCS prediction, as it achieved the best performance in that task. Additionally, we evaluated skipping or including the second stage of the transfer learning training procedure, which consisted of unfreezing the CNN weights and training the whole model using lower learning rates. In this analysis, all models contained a single 1024-dimensional LSTM layer and were trained using sequences of length 1 to 10. The results are shown in Table 4.6. In this case, the best model was achieved when using CORN for classification and initializing the CNN with the weights from the CORAL BCS model while skipping fine-tuning the CNN weights. This shows that the CORAL weights trained in the single-image BCS prediction approach provided a good starting point for the CNN portion of the CNN-RNN model for next-week BCS prediction. Fine-tuning the CNN weights during CNN-RNN training actually hindered performance, possibly due to overfitting, as the ResNet-50 CNN backbone contains over 23 million parameters and the training set, although being in theory virtually infinite because of the large number of possible variations for different frames extracted from each video for building the image sequences, included a total of 839 different videos.

For the next comparative analysis, the number and dimension of LSTM layers were explored using CORN, initializing the CNN with weights from the CORAL model for BCS prediction, and not performing fine-tuning of the CNN weights. The dimensions of the LSTM layers ranged from 64 to 2,048 in powers of two, and we evaluated using one or two LSTM layers as the RNN portion of the model. The best performing model considering 0.25-error accuracy contained a single 512-dimensional LSTM layer, as shown in Table 4.7. This again indicates the presence of overfitting, with deeper or wider models not necessarily translating into better performance.

For the last comparative analysis pertaining to next-week BCS prediction CNN-RNN models, the minimum and maximum image sequence length, as well as whether to include postpartum videos, were evaluated. The other hyperparameters were fixed to their respective optimal values according to previous analyses (CORN, CORAL BCS initial weights, no CNN fine-tuning, and one 512-dimensional LSTM layer). The models were evaluated exclusively on test set sequences containing exactly 3 prepartum frames, as those were the types of sequences that would later be used to extract features from for subclinical ketosis prediction. The best model was achieved by training on all possible sequence lengths, with minimum and maximum lengths set to 1 and 10 and including frames from postpartum videos. These results suggest that increasing the quantity of training data was beneficial for next-week BCS prediction, even when evaluating the models solely on sequences containing three prepartum frames. Detailed results are shown in Table 4.8.

Models for the Early Detection of Subclinical Ketosis

Each of the seven tasks described in the *Subclinical Ketosis Prediction* section resulted in a different comparative analysis: (1) using only depth image features and comparing with a BCS

baseline; (2) combining image features with tabular data; (3) using different number of data points per cow; (4) exploring regression and different BHB thresholds; (5) comparing tabular data with the respective generated text embeddings; (6) including text embeddings from textual notes; and (7) combining tabular data and features from depth images and textual notes to train a final SCK prediction model.

Using Only Depth Image Features

In the first task, SCK prediction models were trained using only features extracted from depth images, comparing the three methods described in the *Image Feature Extraction* section: CNN models for BCS prediction (*CNN Models for BCS Prediction*), body surface between anatomical keypoints (*Body Surface Between Anatomical Keypoints*), and CNN-RNN models for next-week BCS prediction (*CNN-RNN Models for Next-Week BCS Prediction*); as well as a baseline model trained using only the three BCS measurements directly as predictive variables. The best performing model trained on CNN features was achieved using the CORN model features without applying PCA (average $F_1$ score = 0.413; average accuracy = 59.5%). This model was the second best for predicting BCS, as shown in Table 4.5, but was the best when used as a feature extractor for SCK prediction, when compared to using CORAL or regular softmax (0.413 versus 0.329 and 0.352 average $F_1$ score; 59.5% versus 55.0% and 59.2% average accuracy). This indicates that there is some relationship between BCS prediction and the quality of the models as feature extractors for SCK prediction, but it is not an exact correlation. In other words, the best BCS predictor in this case was not necessarily the best feature extractor, especially as the CORAL and CORN frameworks only affect the output layer and loss function utilized during training and do not make changes to the backbone CNN architecture.

The best model trained on body surface depth values between anatomical keypoints was achieved using the lowest sampling resolution of 20 and normalizing the values using the lines that connect each pair of keypoints (average $F_1$ score = 0.455; average accuracy = 57.6%). A lower sampling resolution possibly helped reduce the noise in the depth values, and normalizing the features adjusted for variations that were not necessarily representative of the body shape of the animals, such as body height and inclination at the time that the frames were captured. Overall, the more direct body surface representation achieved using this approach was better at predicting SCK than using the more indirect BCS CNN features.

Using features extracted from next-week BCS prediction CNN-RNNs, the best SCK prediction model was achieved using features from the last LSTM hidden state of the model trained using only sequences of exactly three prepartum frames (average $F_1$ score = 0.493; average accuracy = 63.9%). Although this model was the second worst at predicting next-week BCS, as shown in Table 4.8, it achieved good results for SCK prediction, possibly due to it being focused on prepartum sequences of three frames, which was the same configuration of sequences used for SCK prediction. Additionally, this model was trained for predicting the BCS of the cows close to the calving date (one week after the last prepartum video was collected), which can be more directly related to the risk of SCK (Duffield, 2000) than images and BCS later into lactation. This suggests, again, that the best BCS predictors are not necessarily the best feature extractors for SCK prediction, and training on images taken exclusively during prepartum proved to be beneficial for achieving a good feature extractor for this task.

The baseline model trained using only the three prepartum BCS assessments achieved an average $F_1$ score of 0.473 (average accuracy = 61.8%), which was higher than all except the best model that utilized depth image features. Many of the features extracted from the images might

not be directly related to SCK, thus introducing noise to the model inputs. That is especially relevant for features extracted from deep neural networks that were trained for other tasks (such as BCS), which might not even be directly related to the body fat stores of the cows, but to any other signal captured by the model that facilitates BCS prediction. Conversely, the BCS measurements are more succinct and direct representations of the body shape of the animals, which contain only noise related to the subjectiveness of the human evaluation. However, BCS is a single number that does not incorporate nuances of where the body fat is distributed and how exactly the shape of the body changes across different weeks. Thus, the CNN-RNN approach achieved better results than using only BCS or any other image feature extraction technique explored in this study, possibly because the CNN-RNN model captures relationships between the images of the same cow across different points in time. This means that the CNN-RNN model can extract information from a sequence of prepartum images jointly, as opposed to the other approaches, which rely on extracting features from each image individually and then simply concatenating them. Extracting information from a sequence of images taken across different weeks might be important for SCK prediction, as reflected by the superior predictive ability of the CNN-RNN. Table 4.9 shows a summary of the most relevant models for comparison.

Combining Image Features with Tabular Data

The second comparative analysis consisted of finding the best model that included image features and tabular data. The best image features using each approach (BCS CNN, depth vectors, depth areas, and next-week BCS CNN-RNN) were combined with tabular data, and the results were compared to models that used only tabular data. Additionally, models excluding the two dry matter intake variables were evaluated, as we considered that dry matter intake was the most difficult metric to capture precisely in a commercial farm setting without feed intake control bins.

In contrast, computer vision systems that monitor the other variables related to feeding behavior and cow activity, as well as BCS and body weight, have been more widely explored (Achour et al., 2020; McDonagh et al., 2021; Qiao et al., 2021; Bresolin et al., 2023).

The baseline models containing only tabular data without image features achieved average $F_1$ scores of 0.655 and 0.547 (average accuracies = 74.2% and 67.4%) with and without including DMI, respectively. The best models were achieved by using depth vector features both when including (average $F_1$ score = 0.706; average accuracy = 76.8%) and excluding (average $F_1$ score = 0.596; average accuracy = 72.6%) DMI. When including DMI, the best model used normalized depth vectors with a sampling resolution of 20 without performing PCA, and when excluding DMI, the best model used normalized depth vectors with a sampling resolution of 200 while performing PCA on the image features to convert them to a 23-dimensional space, which was the same number of dimensions as the tabular data when excluding the two DMI variables. The idea behind performing PCA on the image features was to evaluate how reducing the dimensionality, and potentially reducing the noise, of those feature sets affected the results. A summary of the results achieved by the most relevant models evaluated in this comparative analysis is presented in Table 4.10.

The models that included body shape information in a more direct way through extracting depth values between anatomical keypoints performed better than the ones including CNN-RNN features when combining the depth image features with tabular data. Even though the CNN-RNN features were slightly better than depth vectors when used by themselves, the depth vectors include more direct and controlled information about the body shape of the cow presented in each image, which might have benefited the SCK prediction models when coupled with tabular data. Excluding DMI variables greatly harmed the predictive performance of the models, with the best model

achieving an $F_1$ score of 0.596 compared to 0.706 when including DMI. This shows that, although being a relatively difficult variable to achieve in large scale, prepartum DMI can be a great predictor for postpartum SCK. This is further evidenced when analyzing the average feature importances for the best models including or not image features, as illustrated in Figure 4.8. Feature importances were calculated as the mean decrease in impurity (**MDI**) relative to each feature in the RF models, with impurity defined using the Gini criterion.

Using Multiple Data Points per Cow

As shown in Table 4.3, the feature sets extracted from depth images contain potentially $50^3$ variations per cow, as each of the 50 frames extracted from each of the 3 prepartum videos generated a different feature vector. Because of that, for the third comparative analyses we explored randomly sampling 10, 25, and 50 variations of image features per cow, instead of simply calculating the mean feature values for each cow as done in the previous analyses. Using multiple feature vectors per cow, the final prediction for each cow consisted of the most predicted binary class among all samples of that cow. When combining image features with tabular data, the latter was repeated for all data points of the same cow, as each animal contained only a single value for each of the behavior and cow history variables. This analysis was performed using the same sets of image features as the previous analysis, with and without including the DMI variables. The results from the most relevant models are illustrated in Figure 4.9. In general, using multiple samples per cow hindered the performance of the models, except for when using CNN-RNN features. This indicates that simply using the mean values of each feature might be the best approach when using normalized depth vectors, as it possibly provides more stable depth values for each cow, resulting in better predictive performance. In the case of using features extracted from neural networks, calculating mean feature values was detrimental to the performance. Since

the neural networks were simply optimized to perform BCS prediction, there was no guarantee or incentive for images of the same cow to appear close together in the feature space, resulting in mean feature values that might not represent that cow correctly. Instead, using feature vectors for each image or sequence of images individually resulted in better results when using neural network features, as each feature vector extracted now actually represents the corresponding image or sequence of images.

Exploring Regression and Different BHB Thresholds

In the previous comparative analyses, the Random Forest models were trained for binary classification by converting the maximum plasma BHB value measured for each cow between 3 and 14 days in milk to a binary value by using a threshold of 1.0 mmol/L. For the fourth comparative analysis, we explored directly performing regression on the maximum plasma BHB value for each cow and only using the defined SCK threshold for calculating classification performance metrics. Additionally, we evaluated using other threshold values of 1.1 and 1.2 mmol/L for both classification and regression models. The feature sets used to train the regression and classification models were the following: only BCS; only tabular data, including or not DMI; best image features; and best image features combined with tabular data, including or not DMI. Figure 4.10 illustrates the average $F_1$ scores for each model evaluated in this comparative analysis.

The best image features were individually defined according to their performance by themselves or combined with tabular data including or not DMI. The CNN-RNN feature set contained features extracted from the last hidden state of the CNN-RNN for next-week BCS prediction using only sequences containing three prepartum images. When including DMI, the depth vectors were normalized depth values with sampling resolution of 20 without PCA, and when not including DMI the depth vectors were normalized depth values with sampling resolution

of 200 applying PCA with 23 components. The best performing model overall was achieved using tabular data combined with depth vector features for binary classification using a BHB threshold of 1.0 mmol/L (average $F_1$ score = 0.706; average accuracy = 76.8%).

Including image features resulted in a decrease in the $F_1$ score of all models except for when performing a regression and a threshold of 1.1 mmol/L without including DMI, when performing classification with a threshold of 1.2 mmol/L without including DMI, and when performing classification with a threshold of 1.0 mmol/L in all cases. Nevertheless, including image features resulted in the best performing model overall (classification with threshold of 1.0 mmol/L). This indicates that the trained RF models might have struggled to learn the target variable when performing classification with thresholds of 1.1 and 1.2 mmol/L due to imbalance on the number of cows considered as having subclinical ketosis or not. Only 28 and 21 out of 92 cows were considered sick when using thresholds of 1.1 and 1.2 mmol/L respectively, which might have impacted the learning capacity of the models. For the dataset utilized in this study, a threshold of 1.0 mmol/L seems to be the most adequate to use, as 37 out of 92 had maximum plasma BHB measurements above that threshold, resulting in a more balanced dataset and consequently better classification performance overall. When evaluating the recall of the models, including image features resulted in higher values in most cases, as shown in Figure 4.10(b). Recall is an important metric for models that perform early disease detection such as the ones developed in this study, as detecting potential positive cases facilitates early treatment and improves disease prevention. A model that has higher recall and lower precision is preferrable over the opposite, because it is less costly to act on a cow unnecessarily for preventing SCK (false positive) than it is to neglect a cow with a high risk of developing SCK and failing to prevent it (false negative), resulting in higher treatment and indirect costs (Cainzos et al., 2022).

Comparing Tabular Data with Text Embeddings

In the fifth comparative analysis, we compared directly using tabular data as predictors, versus using the text embeddings of the corresponding texts that describe those variables (*template text*), generated via the templates depicted in Table 4.2. The idea behind this analysis was to evaluate whether text embeddings extracted from descriptive text would be able to capture the information included in those variables and be equivalent SCK predictors as using the tabular data directly. When applying or not PCA with 25 components, the models trained using the generated texts achieved average $F_1$ scores of 0.600 and 0.593, respectively (average accuracies of 69.7% and 70.5%), which were lower than the average $F_1$ score achieved by the model trained using the tabular data directly (average $F_1$ score = 0.655; average accuracy = 74.2%). This indicates that the text embeddings extracted from text generated from tabular data using templates are not as good as using the tabular data directly for SCK prediction. Nevertheless, the results were close enough to suggest that some information was captured even using a generic text embedding LLM (OpenAI's *text-embedding-ada-002* model). Since the information contained in the text was very specific to dairy cow management, the generic LLM used might not be ideal for extracting embeddings from the generated text, as it was not trained specifically for this context. The quality of the embeddings might improve if extracted from an LLM fine-tuned using dairy farm management information such as articles written by specialists or scientific papers (Zhu et al., 2023), and further research in this direction would be needed to validate this hypothesis.

The ability to use data in a textual format for phenotype prediction could be a step towards more easily integrating data originating from different sources without the need to strictly follow predetermined data standards. Defining standards for data generated by PLF systems poses a substantial challenge to effective data integration and analysis (Bahlo et al., 2019) in livestock

farms, which could be mitigated if all generated data can be interpreted and analyzed in a common text format. Another potential benefit of using text embeddings extracted from text instead of using structured data as predictors is that systems that capture information in this unstructured format could be used to collect data in dairy farms in a more seamless way, by allowing the farmers to add any type of information about their herd through text or even voice recordings using speech-to-text technology. The data collected from such systems could then be used with predictive models such as the ones proposed in this study, or even be directly requested by user queries using knowledge retrieval natural language processing (**NLP**) techniques (Lewis et al., 2020).

Including Text Embeddings from Textual Notes

In addition to the texts methodically generated from tabular data using templates, notes in the CSV format extracted from the farm management software were converted to natural language text using OpenAI's *GPT-4* chat completion model. This enabled contextual text embeddings to be extracted from this data and used as additional features for SCK prediction. In the sixth comparative analysis for the proposed machine learning pipeline, we explored two methods for including this information contained in textual notes in the predictive models: simply concatenating the 1,536 features extracted from the text (*notes text*) to the 25 tabular data variables; and extracting the text embeddings from a full text description of each cow that combines the textual notes with text generated from tabular data using templates (*combined text*). Using the first approach of concatenating textual notes embeddings to the tabular data resulted in a superior performance when compared to just using the tabular data by itself (average $F_1$ score of 0.681 versus 0.655; average accuracy of 78.7% versus 74.2%). This promising result indicates that the notes recorded in the farm management software including pen moves, vaccine and medicine administration, pregnancy checks, and other information prior to the calving date, contain

important information for SCK prediction. This information is often overlooked due to the difficulty in including this data in quantitative analyses as it is unstructured and often sparse. However, even the generic LLMs utilized in this study, which were not fine-tuned for the dairy farming context, were able to extract relevant embeddings that served as predictors for phenotype modeling. This sheds light on the potential of utilizing data collected in an unstructured way for developing models that aid in the decision-making process in dairy farms and help improve animal health and overall farm management. Table 4.11 provides a summary of the results achieved using text embeddings extracted in different ways for the early detection of SCK.

Combining All Features

The last comparative analysis explored in this study consisted of simply concatenating the best features extracted from each data source (tabular data, depth images, and text notes) and training a model for SCK prediction, comparing its performance with those achieved by previous models. For that, we concatenated the 660 features extracted from depth images using the normalized depth values between anatomical keypoints approach with sampling resolution of 20, the 25 variables collected from wearable sensors and farm management software (tabular data), and the 25 PCA-transformed text embeddings extracted from textual notes. We used one sample per cow, calculating the mean feature value for the image features, and performed binary classification using a plasma BHB threshold of 1.0 mmol/L. The resulting model achieved an average $F_1$ score of 0.680 (average accuracy of 76.1%), which is slightly lower than the best model using only depth images and tabular data (average $F_1$ score = 0.706, average accuracy = 76.8%), and the best model using only notes text and tabular data (average $F_1$ score = 0.681, average accuracy = 78.7%). Since adding image features and text features separately contributed to the predictive performance of the trained models when compared to using only tabular data, we

expected the performance to increase even further when including features extracted from all three sources concurrently. However, it is difficult to draw a permanent conclusion for why that was not the case, and further investigation including a larger number of animals and possibly exploring different feature extraction techniques might be necessary.

Main Implications of Our Findings

The first promising result found in this study was that using image features resulted in better results than using only the BCS values corresponding to the same period when images were taken (average $F_1$ score = 0.493 versus 0.473; average accuracy = 63.9% versus 61.8%). This supports our initial understanding that features extracted from depth images should be more informative of the body shape of the animals than the simple BCS value (or in the case of the CNN-RNN approach, of the body shape variation through time). The image features provide a more comprehensive representation of the cow body shape, holding information about different parts of the body (more explicitly when using depth vectors retrieved between anatomical keypoints), as opposed to the BCS, which is supposed to describe the whole animal condition in a single number. Additionally, image features present a more quantitative way to describe the body shape, as BCS is a partially subjective measurement, with different evaluators potentially having different interpretations of the scoring guidelines.

However, predicting multifaceted health problems such as ketosis (and even more so subclinical ketosis) requires integrating data from multiple different technologies that provide insights about various aspects of the individual cows. Information collected from a single type of sensor (such as cameras) represents only one dimension of the health condition of a cow, which most likely will not be sufficient for a robust understanding and prediction of complex health problems. This explains why using only image features, while performing better than BCS, still

did not achieve impressive results for the early detection of SCK. The predictive performance of the proposed models improved considerably when combing information from multiple different sources, such as wearable sensors that quantify feeding behavior and activity, data logged into the farm management software in the form of structured and unstructured text data, and depth sensing cameras. In general, the inclusion of each data modality (image, tabular data, and text) contributed to the final performance, gradually adding information that enhanced the robustness of the SCK prediction models. Moreover, the fact that the text embeddings extracted from *template text* performed only marginally worse than tabular data (average $F_1$ score = 0.600 versus 0.655; average accuracy = 69.7% versus 74.2%) represents a promising prospect for facilitating data integration for analysis and predictive modeling. Data collected from different PLF systems could be converted to text and the extracted text embeddings could be directly used in quantitative analyses, as opposed to having to define data standards that those systems must conform to in order to enable data analysis (Bahlo et al., 2019).

Some of the data explored in this study had never been considered before for the early detection of health issues in dairy cows, most notably depth images and text data extracted from the farm management software. To the best of our knowledge, this was the first attempt to leverage information contained in this type of data for health monitoring, enabled by the use of modern NLP and computer vision techniques for feature extraction. Furthermore, the proposed machine learning pipeline aims to detect cases of SCK during the postpartum period using exclusively data gathered prepartum, with up to 15 days in advance. Such an early detection system would allow the implementation of management practices that could drastically improve the decision-making process in dairy farms. For example, it could reduce the time that cows spend in the fresh pen, as cows that will most likely not develop hyperketonemia, as predicted by the system, could be moved

to a regular pen earlier. This could reduce costs associated with the time and labor required to check cows in the fresh pen and alleviate overstocking, which is known to cause several issues related to cow welfare and performance (Fregonesi et al., 2007; Coblentz et al., 2018). Additionally, the ability to detect cows with a high risk of developing SCK could lead to the adoption of more focused preventive practices that can reduce the economical impact of SCK in dairy cows (McArt et al., 2014). Future studies evaluating longer prepartum time series could lead to even better dairy management practices if the predictions are good enough, as they could potentially allow an even earlier detection of health issues in individual animals.

Finally, the pipeline proposed in this study could be used for other purposes beyond subclinical ketosis detection. Different phenotypes that are also associated with body tissue mobilization and behavior during the transition period could be predicted using the same features extracted via the proposed data processing methods. This framework could be re-used to perform predictions for other health issues or even other variables related to the transition period, such as reproductive performance, productive potential, and animal welfare.

## CONCLUSION

The main objective of this study was to propose and rigorously evaluate a machine learning pipeline for early subclinical ketosis detection, analyzing different techniques for extracting features from depth images containing the cow body and from textual notes retrieved from the farm management software, and combining the extracted features with tabular data containing feeding behavior, cow activity, and cow history variables collected using wearable sensors and farm management software. This study represented a first attempt at using body shape information extracted from depth images collected through time during prepartum for the early detection of SCK postpartum, leveraging the detailed information contained in those images related to how the

body shape of the cows changed during prepartum, and how they might be useful predictors for early lactation SCK. Additionally, this study also provided a first attempt to integrate textual information extracted from farm management software using NLP for dairy cow phenotype prediction, laying the foundation for further research in this exciting and potentially groundbreaking field. The machine learning pipeline proposed in this study can extract information from high-dimensional unstructured data such as images and text and use it for early disease detection. We believe that the proposed framework can be replicated for the prediction of many other phenotypes that could enhance the decision-making process at dairy farms. The superior results achieved when including depth images of the cows or unstructured textual notes collected via farm management software for SCK prediction shed light on the potential of using this type of data as disease and phenotype predictors. This involves utilizing modern deep learning techniques for extracting quantitative information from this high-dimensional unstructured data that would otherwise be challenging to incorporate in predictive analyses. The detailed exploration of the different ways to extract features from imaging and textual data could lay the foundation for future methods to integrate multiple data sources into robust phenotype prediction models. Furthermore, the proposed automated pipeline could allow the implementation of preventive practices in dairy farms, reducing costs associated with subclinical ketosis, and improving animal health and welfare.

## ACKNOWLEDGMENTS

The authors would like to thank the USDA National Institute of Food and Agriculture (Washington, DC; grant 2023-68014-39821/accession no. 1030367) for the financial support.

## REFERENCES

Achour, B., M. Belkadi, I. Filali, M. Laghrouche, and M. Lahdir. 2020. Image analysis for individual identification and feeding behaviour monitoring of dairy cows based on Convolutional Neural Networks (CNN). Biosyst Eng 198:31–49. doi:https://doi.org/10.1016/j.biosystemseng.2020.07.019.

Andrew, W., C. Greatwood, and T. Burghardt. 2017. Visual Localisation and Individual Identification of Holstein Friesian Cattle via Deep Learning. Pages 2850–2859 in 2017 IEEE International Conference on Computer Vision Workshops (ICCVW).

Bahlo, C., P. Dahlhaus, H. Thompson, and M. Trotter. 2019. The role of interoperable data standards in precision livestock farming in extensive livestock systems: A review. Comput Electron Agric 156:459–466. doi:https://doi.org/10.1016/j.compag.2018.12.007.

Borges Oliveira, D.A., L.G. Ribeiro Pereira, T. Bresolin, R.E. Pontes Ferreira, and J.R. Reboucas Dorea. 2021. A review of deep learning algorithms for computer vision systems in livestock. Livest Sci 253:104700. doi:https://doi.org/10.1016/j.livsci.2021.104700.

Bresolin, T., R. Ferreira, F. Reyes, J. Van Os, and J.R.R. Dórea. 2023. Assessing optimal frequency for image acquisition in computer vision systems developed to monitor feeding behavior of group-housed Holstein heifers. J Dairy Sci 106:664–675. doi:https://doi.org/10.3168/jds.2022-22138.

Buckley, F., K. O'Sullivan, J.F. Mee, R.D. Evans, and P. Dillon. 2003. Relationships Among Milk Yield, Body Condition, Cow Weight, and Reproduction in Spring-Calved Holstein-Friesians. J Dairy Sci 86:2308–2319. doi:https://doi.org/10.3168/jds.S0022-0302(03)73823-5.

Caffarini, J.G., T. Bresolin, and J.R.R. Dorea. 2022. Predicting ribeye area and circularity in live calves through 3D image analyses of body surface. J Anim Sci 100:skac242. doi:10.1093/jas/skac242.

Cainzos, J.M., C. Andreu-Vazquez, M. Guadagnini, A. Rijpert-Duvivier, and T. Duffield. 2022. A systematic review of the cost of ketosis in dairy cattle. J Dairy Sci 105:6175–6195. doi:https://doi.org/10.3168/jds.2021-21539.

Cameron, R.E.B., P.B. Dyk, T.H. Herdt, J.B. Kaneene, R. Miller, H.F. Bucholtz, J.S. Liesman, M.J. Vandehaar, and R.S. Emery. 1998. Dry Cow Diet, Management, and Energy Balance as Risk Factors for Displaced Abomasum in High Producing Dairy Herds. J Dairy Sci 81:132–139. doi:https://doi.org/10.3168/jds.S0022-0302(98)75560-2.

Cao, W., V. Mirjalili, and S. Raschka. 2020. Rank consistent ordinal regression for neural networks with application to age estimation. Pattern Recognit Lett 140:325–331. doi:https://doi.org/10.1016/j.patrec.2020.11.008.

Cardoso, F.C., K.F. Kalscheur, and J.K. Drackley. 2020. Symposium review: Nutrition strategies for improved health, production, and fertility during the transition period. J Dairy Sci 103:5684–5693. doi:https://doi.org/10.3168/jds.2019-17271.

Coblentz, W.K., M.S. Akins, N.M. Esser, R.K. Ogden, and S.L. Gelsinger. 2018. Effects of overstocking at the feedbunk on the growth performance and sorting characteristics of a

forage-based diet offered for ad libitum intake to replacement Holstein dairy heifers. J Dairy Sci 101:7930–7941. doi:https://doi.org/10.3168/jds.2018-14543.

Cominotte, A., A.F.A. Fernandes, J.R.R. Dorea, G.J.M. Rosa, M.M. Ladeira, E.H.C.B. van Cleef, G.L. Pereira, W.A. Baldassini, and O.R. Machado Neto. 2020. Automated computer vision system to predict body weight and average daily gain in beef cattle during growing and finishing phases. Livest Sci 232:103904. doi:https://doi.org/10.1016/j.livsci.2019.103904.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. Pages 248–255 in 2009 IEEE Conference on Computer Vision and Pattern Recognition.

Dórea, J.R.R., E.A. French, and L.E. Armentano. 2017. Use of milk fatty acids to estimate plasma nonesterified fatty acid concentrations as an indicator of animal energy balance. J Dairy Sci 100:6164–6176. doi:https://doi.org/10.3168/jds.2016-12466.

Drackley, J.K. 1999. Biology of Dairy Cows During the Transition Period: the Final Frontier?. J Dairy Sci 82:2259–2273. doi:https://doi.org/10.3168/jds.S0022-0302(99)75474-3.

Dubuc, J., T.F. Duffield, K.E. Leslie, J.S. Walton, and S.J. LeBlanc. 2010. Definitions and diagnosis of postpartum endometritis in dairy cows. J Dairy Sci 93:5225–5233. doi:https://doi.org/10.3168/jds.2010-3428.

Duffield, T. 2000. Subclinical Ketosis in Lactating Dairy Cattle. Veterinary Clinics of North America: Food Animal Practice 16:231–253. doi:https://doi.org/10.1016/S0749-0720(15)30103-1.

Enjalbert, F., M.C. Nicot, C. Bayourthe, and R. Moncoulon. 2001. Ketone Bodies in Milk and Blood of Dairy Cows: Relationship between Concentrations and Utilization for Detection of Subclinical Ketosis. J Dairy Sci 84:583–589. doi:https://doi.org/10.3168/jds.S0022-0302(01)74511-0.

Fernandes, A.F.A., J.R.R. Dórea, R. Fitzgerald, W. Herring, and G.J.M. Rosa. 2019. A novel automated system to acquire biometric and morphological measurements and predict body weight of pigs via 3D computer vision. J Anim Sci 97:496–508. doi:10.1093/jas/sky418.

Fernandes, A.F.A., J.R.R. Dórea, B.D. Valente, R. Fitzgerald, W. Herring, and G.J.M. Rosa. 2020. Comparison of data analytics strategies in computer vision systems to predict pig body composition traits from 3D images. J Anim Sci 98:skaa250. doi:10.1093/jas/skaa250.

Fregonesi, J.A., C.B. Tucker, and D.M. Weary. 2007. Overstocking Reduces Lying Time in Dairy Cows. J Dairy Sci 90:3349–3354. doi:https://doi.org/10.3168/jds.2006-794.

Goldhawk, C., N. Chapinal, D.M. Veira, D.M. Weary, and M.A.G. von Keyserlingk. 2009. Prepartum feeding behavior is an early indicator of subclinical ketosis. J Dairy Sci 92:4971–4977. doi:https://doi.org/10.3168/jds.2009-2242.

González, L.A., B.J. Tolkamp, M.P. Coffey, A. Ferret, and I. Kyriazakis. 2008. Changes in Feeding Behavior as Possible Indicators for the Automatic Monitoring of Health Disorders in Dairy Cows. J Dairy Sci 91:1017–1028. doi:https://doi.org/10.3168/jds.2007-0530.

Green, B.L., B.W. McBride, D. Sandals, K.E. Leslie, R. Bagg, and P. Dick. 1999. The Impact of a Monensin Controlled-Release Capsule on Subclinical Ketosis in the Transition Dairy Cow. J Dairy Sci 82:333–342. doi:https://doi.org/10.3168/jds.S0022-0302(99)75240-9.

Grummer, R.R. 1993. Etiology of Lipid-Related Metabolic Disorders in Periparturient Dairy Cows. J Dairy Sci 76:3882–3896. doi:https://doi.org/10.3168/jds.S0022-0302(93)77729-2.

Hammon, D.S., I.M. Evjen, T.R. Dhiman, J.P. Goff, and J.L. Walters. 2006. Neutrophil function and energy status in Holstein cows with uterine health disorders. Vet Immunol Immunopathol 113:21–29. doi:https://doi.org/10.1016/j.vetimm.2006.03.022.

He, K., X. Zhang, S. Ren, and J. Sun. 2016. Deep residual learning for image recognition. Pages 770–778 in Proceedings of the IEEE conference on computer vision and pattern recognition.

Hochreiter, S., and J. Schmidhuber. 1997. Long Short-Term Memory. Neural Comput 9:1735–1780. doi:10.1162/neco.1997.9.8.1735.

Holdorf, H.T., S.J. Kendall, K.E. Ruh, M.J. Caputo, G.J. Combs, S.J. Henisz, W.E. Brown, T. Bresolin, R.E.P. Ferreira, J.R.R. Dorea, and H.M. White. 2023. Increasing the prepartum dose of rumen-protected choline: Effects on milk production and metabolism in high-producing Holstein dairy cows. J Dairy Sci 106:5988–6004. doi:https://doi.org/10.3168/jds.2022-22905.

Horst, R.L., J.P. Goff, and T.A. Reinhardt. 1994. Calcium and Vitamin D Metabolism in the Dairy Cow. J Dairy Sci 77:1936–1951. doi:https://doi.org/10.3168/jds.S0022-0302(94)77140-X.

Itle, A.J., J.M. Huzzey, D.M. Weary, and M.A.G. von Keyserlingk. 2015. Clinical ketosis and standing behavior in transition cows. J Dairy Sci 98:128–134. doi:https://doi.org/10.3168/jds.2014-7932.

Jocher, G., A. Chaurasia, and J. Qiu. 2023. YOLO by Ultralytics (Version 8.0. 0)[Computer software]. YOLO by Ultralytics (Version 8.0. 0)[Computer software].

Jorjong, S., A.T.M. van Knegsel, J. Verwaeren, M.V. Lahoz, R.M. Bruckmaier, B. De Baets, B. Kemp, and V. Fievez. 2014. Milk fatty acids as possible biomarkers to early diagnose elevated concentrations of blood plasma nonesterified fatty acids in dairy cows. J Dairy Sci 97:7054–7064. doi:https://doi.org/10.3168/jds.2014-8039.

Keselman, L., J.I. Woodfill, A. Grunnet-Jepsen, and A. Bhowmik. 2017. Intel(R) RealSense(TM) Stereoscopic Depth Cameras. Pages 1267–1276 in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).

Kingma, D.P., and J. Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

LeBlanc, S. 2010. Monitoring Metabolic Health of Dairy Cattle in the Transition Period. Journal of Reproduction and Development 56:S29–S35. doi:10.1262/jrd.1056S29.

LeBlanc, S.J., K.E. Leslie, and T.F. Duffield. 2005. Metabolic Predictors of Displaced Abomasum in Dairy Cattle. J Dairy Sci 88:159–170. doi:https://doi.org/10.3168/jds.S0022-0302(05)72674-6.

Lei, M.A.C., and J. Simões. 2021. Invited Review: Ketosis Diagnosis and Monitoring in High-Producing Dairy Cows. Dairy 2:303–325. doi:10.3390/dairy2020025.

Lewis, P., E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Pages 9459–9474 in Advances in Neural Information Processing Systems. Curran Associates, Inc.

Liu, D., D. He, and T. Norton. 2020. Automatic estimation of dairy cattle body condition score from depth image using ensemble model. Biosyst Eng 194:16–27. doi:https://doi.org/10.1016/j.biosystemseng.2020.03.011.

McArt, J.A.A., D. V Nydam, G.R. Oetzel, and C.L. Guard. 2014. An economic analysis of hyperketonemia testing and propylene glycol treatment strategies in early lactation dairy cattle. Prev Vet Med 117:170–179. doi:https://doi.org/10.1016/j.prevetmed.2014.06.017.

McDonagh, J., G. Tzimiropoulos, K.R. Slinger, Z.J. Huggett, P.M. Down, and M.J. Bell. 2021. Detecting Dairy Cow Behavior Using Vision Technology. Agriculture 11. doi:10.3390/agriculture11070675.

Menezes, G.L., T. Bresolin, R. Ferreira, H.T. Holdorf, S.I. Arriola Apelo, H.M. White, and JoaoR.R. Dórea. 2024. Near-infrared spectroscopy analysis of blood plasma for predicting nonesterified fatty acid concentrations in dairy cows. JDS Communications 5:195–199. doi:https://doi.org/10.3168/jdsc.2023-0458.

Min, B., H. Ross, E. Sulem, A.P. Ben Veyseh, T.H. Nguyen, O. Sainz, E. Agirre, I. Heintz, and D. Roth. 2023. Recent Advances in Natural Language Processing via Large Pre-trained Language Models: A Survey. ACM Comput. Surv. 56. doi:10.1145/3605943.

Overton, T.R., and M.R. Waldron. 2004. Nutritional Management of Transition Dairy Cows: Strategies to Optimize Metabolic Health. J Dairy Sci 87:E105–E119. doi:https://doi.org/10.3168/jds.S0022-0302(04)70066-1.

Qiao, Y., H. Kong, C. Clark, S. Lomax, D. Su, S. Eiffert, and S. Sukkarieh. 2021. Intelligent perception for cattle monitoring: A review for cattle identification, body condition score evaluation, and weight estimation. Comput Electron Agric 185:106143. doi:https://doi.org/10.1016/j.compag.2021.106143.

Qiao, Y., D. Su, H. Kong, S. Sukkarieh, S. Lomax, and C. Clark. 2019. Individual Cattle Identification Using a Deep Learning Based Framework. IFAC-PapersOnLine 52:318–323. doi:https://doi.org/10.1016/j.ifacol.2019.12.558.

Roche, J.R., N.C. Friggens, J.K. Kay, M.W. Fisher, K.J. Stafford, and D.P. Berry. 2009. Invited review: Body condition score and its association with dairy cow productivity, health, and welfare. J Dairy Sci 92:5769–5801. doi:https://doi.org/10.3168/jds.2009-2431.

Roche, J.R., S. Meier, A. Heiser, M.D. Mitchell, C.G. Walker, M.A. Crookenden, M.V. Riboni, J.J. Loor, and J.K. Kay. 2015. Effects of precalving body condition score and prepartum feeding level on production, reproduction, and health parameters in pasture-based transition dairy cows. J Dairy Sci 98:7164–7182. doi:https://doi.org/10.3168/jds.2014-9269.

Ronneberger, O., P. Fischer, and T. Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. Pages 234–241 in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Springer International Publishing, Cham.

de Roos, A.P.W., H.J.C.M. van den Bijgaart, J. Hørlyk, and G. de Jong. 2007. Screening for Subclinical Ketosis in Dairy Cattle by Fourier Transform Infrared Spectrometry. J Dairy Sci 90:1761–1766. doi:https://doi.org/10.3168/jds.2006-203.

Shi, X., W. Cao, and S. Raschka. 2023. Deep neural networks for rank-consistent ordinal regression based on conditional probabilities. Pattern Analysis and Applications 26:941–955. doi:10.1007/s10044-023-01181-9.

Song, X., E.A.M. Bokkers, S. van Mourik, P.W.G. Groot Koerkamp, and P.P.J. van der Tol. 2019. Automated body condition scoring of dairy cows using 3-dimensional feature extraction from multiple body regions. J Dairy Sci 102:4294–4308. doi:https://doi.org/10.3168/jds.2018-15238.

Steeneveld, W., P. Amuta, F.J.S. van Soest, R. Jorritsma, and H. Hogeveen. 2020. Estimating the combined costs of clinical and subclinical ketosis in dairy cows. PLoS One 15:e0230448-.

Wildman, E.E., G.M. Jones, P.E. Wagner, R.L. Boman, H.F. Troutt, and T.N. Lesch. 1982. A Dairy Cow Body Condition Scoring System and Its Relationship to Selected Production Characteristics. J Dairy Sci 65:495–501. doi:https://doi.org/10.3168/jds.S0022-0302(82)82223-6.

Wilson, D.J., and G.M. Goodell. 2013. Comparison of blood strips, milk strips and automated milk measurement of beta-hydroxybutyrate in periparturient dairy cattle and resultant diagnoses of ketosis.

Yukun, S., H. Pengju, W. Yujie, C. Ziqi, L. Yang, D. Baisheng, L. Runze, and Z. Yonggen. 2019. Automatic monitoring system for individual dairy cows based on a deep learning framework that provides identification via body parts and estimation of body condition score. J Dairy Sci 102:10140–10151. doi:https://doi.org/10.3168/jds.2018-16164.

Zhu, J., R. Lacroix, and K.M. Wade. 2023. Automated extraction of domain knowledge in the dairy industry. Comput Electron Agric 214:108330. doi:https://doi.org/10.1016/j.compag.2023.108330.

Zin, T.T., P.T. Seint, P. Tin, Y. Horii, and I. Kobayashi. 2020. Body Condition Score Estimation Based on Regression Analysis Using a 3D Camera. Sensors 20. doi:10.3390/s20133705.

## TABLES AND FIGURES

**Table 4.1.** Hyperparameters explored for training different BCS CNN models and next-week BCS CNN-RNN models.

| Approach | Hyperparameter | Possible values |
|---|---|---|
| **BCS prediction CNN** | Training paradigm | Regular softmax, CORAL, or CORN |
| **Next-week BCS prediction CNN-RNN** | Training paradigm | Regular softmax, CORAL, or CORN |
| | Initial CNN weights | ImageNet or best BCS CNN |
| | Skip second stage of transfer learning? | Yes or no |
| | LSTM layers | 1 or 2 |
| | LSTM dimension | 64, 128, 256, 512, 1,024, or 2,048 |
| | Minimum sequence length | 1, 3, or 5 |
| | Maximum sequence length | 2, 5, or 10 |
| | Include postpartum? | Yes or no |

**Table 4.2.** Templates utilized for converting the feeding behavior, cow activity, and cow history variables into text. The variable values were inserted into the templates in their corresponding position (in {*italic}*), and text embeddings were extracted from each generated text using the *text-embedding-ada-002* model, resulting in a 1,536-dimensional feature vector for each text. The parity variable was converted to its corresponding ordinal text (second, third, fourth, fifth, and sixth).

| |
|---|
| It is a *{Parity}* lactation cow. Its previous lactation lasted *{Previous DIM}* days. Between the previous and current lactations, it stayed *{Previous days dry}* days dry. It had *{Ketosis events}* previous cases of ketosis. Its average daily dry matter intake was *{Intake -7}* kg during the last seven days prepartum and *{Intake -2}* kg during the last two days prepartum. Its average daily time spent eating was *{Feeding time -7}* minutes during the last seven days prepartum and *{Feeding time -2}* minutes during the last two days prepartum. Its average meal duration was *{Meal duration -7}* minutes during the last seven days prepartum and *{Meal duration -2}* minutes during the last two days prepartum. Its average daily number of meals was *{Number of meals -7}* during the last seven days prepartum and *{Number of meals -2}* during the last two days prepartum. Its average daily time spent lying, ruminating, inactive, and highly active in the last seven days prepartum was *{Lying time -7}*, *{Rumination time -7}*, *{Inactive -7}*, and *{Highly active -7}* minutes, respectively. Its body condition score on 21, 14, and 7 days prepartum was *{BCS -21}*, *{BCS -14}*, and *{BCS -7}*, respectively. Its body weight on 21, 14, and 7 days prepartum was *{Body weight -21}*, *{Body weight -14}*, and *{Body weight -7}*, respectively. |
| The cow is on its *{Parity}* lactation. Its prior lactation endured a span of *{Previous DIM}* days. It experienced a dry period of *{Previous days dry}* days between the previous and current lactations. It encountered *{Ketosis events}* occurrences of ketosis previously. Its average daily intake of dry matter registered an amount of *{Intake -7}* kg during the seven days leading up to parturition and *{Intake -2}* kg during the final two days before parturition. Its typical daily feeding duration measured an average of *{Feeding time -7}* minutes during the last seven days prepartum and *{Feeding time -2}* minutes during the last two days prepartum. Its meal duration measured an average of *{Meal duration -7}* minutes during the last seven days prepartum and *{Meal duration -2}* minutes during the last two days prepartum. Its daily number of meals measured an average of *{Number of meals -7}* during the last seven days prepartum and *{Number of meals -2}* during the last two days prepartum. In terms of rest and activity, its daily periods spent lying, ruminating, being inactive, and highly active during the week before calving were *{Lying time -7}*, *{Rumination time -7}*, *{Inactive -7}*, and *{Highly active -7}* minutes, respectively. Its body condition score was *{BCS -21}* on 21 days prepartum, *{BCS -14}* on 14 days prepartum, and *{BCS -7}* on 7 days prepartum. Its body weight was *{Body weight -21}* on 21 days prepartum, *{Body weight -14}* on 14 days prepartum, and *{Body weight -7}* on 7 days prepartum. |
| A cow in the *{Parity}* lactation phase. Its preceding lactation spanned *{Previous DIM}* days. It experienced *{Previous days dry}* dry days between the preceding and current lactations. There were *{Ketosis events}* prior instances of ketosis. The average daily dry matter intake over the last seven days before calving was *{Intake -7}* kg, and *{Intake -2}* kg during the last two days prepartum. It spent an average of *{Feeding time -7}* minutes eating daily during the last seven days prepartum and *{Feeding time -2}* minutes during the last two days prepartum. The typical meal duration was *{Meal duration -7}* minutes during the last seven days prepartum and *{Meal duration -2}* minutes during the last two days prepartum. The typical daily number of meals was *{Number of meals -7}* during the last seven days prepartum and *{Number of meals -2}* |

| |
|---|
| during the last two days prepartum. In the last seven days prepartum, it spent *{Lying time -7}* minutes lying, *{Rumination time -7}* minutes ruminating, *{Inactive -7}* minutes inactive, and *{Highly active -7}* minutes highly active daily. Body condition score on days 21, 14, and 7 prepartum was *{BCS -21}*, *{BCS -14}*, and *{BCS -7}*, respectively. Body weight on days 21, 14, and 7 prepartum was *{Body weight -21}*, *{Body weight -14}*, and *{Body weight -7}*, respectively. |
| A cow is currently in the *{Parity}* lactation phase, with its previous lactation lasting *{Previous DIM}* days. It experienced a dry period of *{Previous days dry}* days between the previous and current lactations. The cow has a history of *{Ketosis events}* previous cases of ketosis. In the last seven days prepartum, it had an average daily dry matter intake of *{Intake -7}* kg, which changed to *{Intake -2}* kg during the last two days prepartum. The cow spent an average of *{Feeding time -7}* minutes eating daily over the last seven days and *{Feeding time -2}* minutes during the last two days prepartum, with meal durations of *{Meal duration -7}* minutes over the last seven days and *{Meal duration -2}* minutes during the last two days prepartum. In the last seven days prepartum, it had a daily average number of meals of *{Number of meals -7}*, which changed to *{Number of meals -2}* during the last two days prepartum. Additionally, in the last seven days prepartum, the cow spent *{Lying time -7}* minutes lying, *{Rumination time -7}* minutes ruminating, *{Inactive -7}* minutes inactive, and *{Highly active -7}* minutes highly active daily. Its body condition score on 21, 14, and 7 days prepartum was *{BCS -21}*, *{BCS -14}*, and *{BCS -7}*, respectively. Its body weight on 21, 14, and 7 days prepartum was *{Body weight -21}*, *{Body weight -14}*, and *{Body weight -7}*, respectively. |
| A cow in its *{Parity}* lactation phase. It previously underwent a lactation period lasting *{Previous DIM}* days. Following the previous lactation and preceding the current one, it remained dry for *{Previous days dry}* days. It experienced *{Ketosis events}* instances of ketosis in the past. The average daily dry matter intake stood at *{Intake -7}* kg over the last seven days prepartum and *{Intake -2}* during the final two days prepartum. Its average daily time dedicated to eating was *{Feeding time -7}* minutes within the last seven days prepartum and *{Feeding time -2}* minutes during the last two days prepartum. The average duration of its meals was *{Meal duration -7}* minutes over the last seven days prepartum and *{Meal duration -2}* minutes during the last two days prepartum. The average daily number of meals was *{Number of meals -7}* over the last seven days prepartum and *{Number of meals -2}* during the last two days prepartum. During the last seven days prepartum, it spent *{Lying time -7}* minutes lying down, *{Rumination time -7}* minutes ruminating, *{Inactive -7}* minutes inactive, and *{Highly active -7}* minutes highly active daily. Its body condition score on days 21, 14, and 7 prepartum was *{BCS -21}*, *{BCS -14}*, and *{BCS -7}*, respectively. Its body weight on days 21, 14, and 7 prepartum was *{Body weight -21}*, *{Body weight -14}*, and *{Body weight -7}*, respectively. |

**Table 4.3.** Description of the features extracted for each cow, generated from data originating from imaging and wearable sensors, and cow history and textual notes collected from farm management software.

| Source | Feature set (number) | Variations per cow | Description |
|---|---|---|---|
| **Depth images** | CNN for BCS (6,144) | $50^3$ | 2,048 features extracted from the second-to-last layer of a CNN for BCS prediction, for each of the 50 depth frames sampled from each video. Features from one frame of each prepartum video were concatenated into a 6,144-sized feature vector for that cow. |
| | Depth vectors (660 to 6,600) | $50^3$ | 220 to 2,200 (depending on sampling resolution) depth values sampled between keypoints on the cow body surface for each depth frame. Values from one frame of each prepartum video were concatenated into a 660- to 6,600-sized feature vector for that cow. Normalized versions of the depth values were also considered. |
| | Areas between keypoints (33) | $50^3$ | 11 pairs of keypoints generated 11 areas that were calculated between the sampled depth vectors and the line connecting two keypoints on the cow body surface for each depth frame. Values from one frame of each prepartum video were concatenated into a 33-sized feature vector for that cow. |
| | CNN-RNN for BCS, per image (192 to 6,144) | $50^3$ | 64 to 2,048 features extracted from the hidden states of the last LSTM layer for each of the three images of the sequence (one for each prepartum video), which were then concatenated into a 192- to 6,144-sized feature vector. |
| | CNN-RNN for BCS, per sequence (64 to 2,048) | $50^3$ | 64 to 2,048 features extracted from the last hidden state of the last LSTM layer for the whole sequence of three images of the sequence (one for each prepartum video. |
| **Wearable sensors** | Intake -7 (1) | 1 | Average daily dry matter intake during the last 7 days prior to calving. |
| | Intake -2 (1) | 1 | Average daily dry matter intake during the last 2 days prior to calving. |
| | Feeding time -7 (1) | 1 | Average daily time spent feeding during the last 7 days prior to calving. |
| | Feeding time -2 (1) | 1 | Average daily time spent feeding during the last 2 days prior to calving. |
| | Meal duration -7 (1) | 1 | Average meal duration during the last 7 days prior to calving. |
| | Meal duration -2 (1) | 1 | Average meal duration during the last 2 days prior to calving. |
| | Number of meals -7 (1) | 1 | Average daily number of meals during the last 7 days prior to calving. |
| | Number of meals -2 (1) | 1 | Average daily number of meals during the last 2 days prior to calving. |
| | Lying time -7 (1) | 1 | Average daily time spent lying during the last 7 days prior to calving. |
| | Rumination time -7 (1) | 1 | Average daily time spent ruminating during the last 7 days prior to calving. |
| | Inactive -7 (1) | 1 | Average daily time spent inactive during the last 7 days prior to calving. |
| | Highly active -7 (1) | 1 | Average daily time spent highly active during the last 7 days prior to calving. |
| **Management software** | Parity dummy variables (4) | 1 | 4 one-hot encoded dummy variables representing cow parity. Second lactation cows were encoded as 0000, third lactation as 0001, fourth lactation as 0010, fifth lactation as 0100, and sixth lactation as 1000. |
| | Previous days in milk (1) | 1 | Number of days in milk in the previous lactation. |

| Source | Feature set (number) | Variations per cow | Description |
|---|---|---|---|
| **Management Software** | Previous days dry (1) | 1 | Number of days dry between previous and current lactations. |
| | Ketosis events (1) | 1 | Total number of ketosis events in previous lactations. |
| | BCS -21 (1) | 1 | BCS assessed 21 days before expected calving date. |
| | BCS -14 (1) | 1 | BCS assessed 14 days before expected calving date. |
| | BCS -7 (1) | 1 | BCS assessed 7 days before expected calving date. |
| | Body weight -21 (1) | 1 | Body weight measured 21 days before expected calving date. |
| | Body weight -14 (1) | 1 | Body weight measured 14 days before expected calving date. |
| | Body weight -7 (1) | 1 | Body weight measured 7 days before expected calving date. |
| **Text** | Template text (1,536) | 5 | 1,536 features extracted from text generated from the wearable sensors and management software variables using 5 templates. |
| | Notes text (1,536) | 1 | 1,536 features extracted from the textual notes retrieved from the management software. |
| | Combined text (1,536) | 5 | 1,536 features extracted from text combining templates and textual notes. |

**Table 4.4.** Hyperparameters optimized during the training of the Random Forest models. The best set of hyperparameters in each train-test split iteration was found via 5-fold cross-validation on the training cows, among 100 random combinations of hyperparameters (randomized search).

| Hyperparameter | Description | Possible values |
|---|---|---|
| **Bootstrap** | Whether to bootstrap samples when building trees (True) or use the whole dataset (False). | *True* or *False* |
| **Maximum depth** | Maximum depth of the trees. If *None*, nodes are expanded until all leaves are pure, or until all leaves contain fewer than *Minimum samples to split* samples. | 10 to 100, increasing by increments of 10, or *None* |
| **Maximum features** | Maximum number of features to consider when looking for the best split. | The total number of features, the logarithm base 2 of the number of features, or the square root of the number of features |
| **Minimum samples for leaf** | Minimum number of samples required to be at a leaf node. | 1, 2, or 4 |
| **Minimum samples to split** | Minimum number of samples required to split an internal node. | 2, 5, or 10 |
| **Number of estimators** | Number of estimators (trees) in the random forest. | 200 to 2000, increasing by increments of 200 |

**Table 4.5.** Results of the single-image BCS prediction models. The best results for each error tolerance are highlighted in bold.

| Method | Accuracy with error tolerance | | | | |
|---|---|---|---|---|---|
| | **0** | **0.25** | **0.50** | **0.75** | **1.0** |
| Regular softmax | 28.4% | 66.9% | 86.6% | 96.0% | 98.4% |
| **CORAL** | **33.4%** | **78.9%** | **94.2%** | **98.9%** | **99.9%** |
| CORN | 31.5% | 73.3% | 90.4% | 97.7% | 99.3% |

**Table 4.6.** Results of the CNN-RNN comparative analysis exploring different training paradigms, initial CNN weights, and whether to fine-tune the CNN weights during training. The best results for each metric, as well as the model that achieved the highest 0.25-error accuracy, are highlighted in bold.

| Method | Initial CNN weights | Fine-tune CNN? | LSTM layers | LSTM dimension | 0-error accuracy | 0.25-error accuracy | 0.50-error accuracy |
|---|---|---|---|---|---|---|---|
| Softmax | ImageNet | Yes | 1 | 1,024 | 32.8% | 75.4% | 95.7% |
| Softmax | CORAL BCS | Yes | 1 | 1,024 | 32.0% | 79.7% | 95.1% |
| Softmax | CORAL BCS | No | 1 | 1,024 | **34.4%** | 79.7% | 96.5% |
| CORAL | ImageNet | Yes | 1 | 1,024 | 29.1% | 72.9% | 91.7% |
| CORAL | CORAL BCS | Yes | 1 | 1,024 | 29.7% | 73.8% | 93.6% |
| CORAL | CORAL BCS | No | 1 | 1,024 | 32.2% | 76.1% | 94.0% |
| CORN | ImageNet | Yes | 1 | 1,024 | 30.7% | 77.6% | 93.9% |
| CORN | CORAL BCS | Yes | 1 | 1,024 | 33.6% | 80.4% | **96.8%** |
| **CORN** | **CORAL BCS** | **No** | **1** | **1,024** | 34.0% | **81.3%** | 96.7% |

**Table 4.7.** Results of the CNN-RNN comparative analysis exploring different number of LSTM layers and their dimensions. The best results for each metric, as well as the model that achieved the highest 0.25-error accuracy, are highlighted in bold.

| Method | Initial CNN weights | Fine-tune CNN? | LSTM layers | LSTM dimension | 0-error accuracy | 0.25-error accuracy | 0.50-error accuracy |
|---|---|---|---|---|---|---|---|
| CORN | CORAL BCS | No | 64 | 1 | 34.3% | 80.7% | **97.9%** |
| CORN | CORAL BCS | No | 128 | 1 | 34.8% | 80.6% | 97.5% |
| CORN | CORAL BCS | No | 256 | 1 | 35.4% | 81.4% | 96.7% |
| **CORN** | **CORAL BCS** | **No** | **512** | **1** | 34.7% | **82.7%** | 97.0% |
| CORN | CORAL BCS | No | 1,024 | 1 | 34.0% | 81.3% | 96.7% |
| CORN | CORAL BCS | No | 2,048 | 1 | 33.4% | 79.9% | 96.9% |
| CORN | CORAL BCS | No | 64 | 2 | **37.1%** | 81.8% | 96.6% |
| CORN | CORAL BCS | No | 128 | 2 | 33.7% | 82.3% | 97.0% |
| CORN | CORAL BCS | No | 256 | 2 | 35.5% | 81.7% | 96.9% |
| CORN | CORAL BCS | No | 512 | 2 | 34.4% | 80.3% | 97.1% |
| CORN | CORAL BCS | No | 1,024 | 2 | 33.9% | 82.6% | 96.8% |
| CORN | CORAL BCS | No | 2,048 | 2 | 35.7% | 82.4% | 97.6% |

**Table 4.8.** Results of the CNN-RNN comparative analysis exploring different sequence lengths and whether to include postpartum frames. The number of training sequences shown in this table was calculated considering one frame per video, but the models were trained using random frame combinations for each video sequence within an epoch, resulting in a virtual infinite number of different data points. The best results for each metric, as well as the model that achieved the highest 0.25-error accuracy, are highlighted in bold. Testing set containing only sequences of 3 prepartum frames.

| Minimum sequence length | Maximum sequence length | Include postpartum? | Number of training sequences | 0-error accuracy | 0.25-error accuracy | 0.50-error accuracy |
|---|---|---|---|---|---|---|
| 3 | 3 | No | 92 | 25.5% | 61.9% | 85.0% |
| 1 | 3 | No | 580 | **33.1%** | 63.3% | 87.9% |
| 3 | 3 | Yes | 828 | 23.8% | 76.2% | 91.7% |
| 5 | 5 | Yes | 614 | 31.7% | 53.3% | 85.0% |
| 3 | 10 | Yes | 3,652 | 17.6% | 73.8% | 94.5% |
| **1** | **10** | **Yes** | **5,638** | 19.5% | **77.9%** | **95.5%** |

**Table 4.9.** Results of SCK prediction models trained using only depth image features or BCS. SR stands for sampling resolution, MinSeq and MaxSeq are the minimum and maximum sequence lengths for training the CNN-RNN models, and "all HL" and "last HL" correspond to using as features all or only the last hidden state of the LSTM layer of the CNN-RNN models. No PCA was applied for any of the models listed in this table. The $F_1$ scores and accuracies are reported as (mean ± standard deviation) across 20 random iterations of training and testing splits. The best results for each metric, as well as the model that achieved the highest $F_1$ score for SCK prediction, are highlighted in bold. The CNN-RNN approach achieved the best performance, indicating that it might be able to extract more relevant information from all prepartum images jointly, as opposed to the other approaches, which rely on extracting features from each image individually.

| Approach | Details | Number of features | $F_1$ score (mean ± SD) | Accuracy % (mean ± SD) |
|---|---|---|---|---|
| BCS only (baseline) | | 3 | 0.473 ± 0.159 | 61.8 ± 9.1 |
| BCS CNN | Softmax | 6,144 | 0.352 ± 0.176 | 59.2 ± 11.9 |
| BCS CNN | CORAL | 6,144 | 0.329 ± 0.160 | 55.0 ± 9.9 |
| BCS CNN | CORN | 6,144 | 0.413 ± 0.182 | 59.5 ± 10.0 |
| Depth vectors | SR=20 | 660 | 0.329 ± 0.139 | 48.7 ± 8.0 |
| Depth vectors | SR=20, normalized | 660 | 0.455 ± 0.153 | 57.6 ± 9.9 |
| Depth vectors | SR=50, normalized | 1,650 | 0.388 ± 0.173 | 55.5 ± 11.6 |
| Depth vectors | SR=100, normalized | 3,300 | 0.392 ± 0.171 | 54.2 ± 8.8 |
| Depth vectors | SR=200, normalized | 6,600 | 0.397 ± 0.201 | 56.6 ± 13.0 |
| Depth areas | SR=200, normalized | 33 | 0.432 ± 0.166 | 58.4 ± 10.5 |
| BCS CNN-RNN | MinSeq=3, MaxSeq=3, prepartum, all HL | 1,536 | 0.441 ± 0.135 | 60.0 ± 10.0 |
| **BCS CNN-RNN** | **MinSeq=3, MaxSeq=3, prepartum, last HL** | **512** | **0.493 ± 0.110** | **63.9 ± 8.2** |
| BCS CNN-RNN | MinSeq=1, MaxSeq=3, prepartum, last HL | 512 | 0.433 ± 0.109 | 58.9 ± 8.7 |
| BCS CNN-RNN | MinSeq=3, MaxSeq=3, pre + postpartum, last HL | 512 | 0.443 ± 0.141 | 60.5 ± 10.1 |
| BCS CNN-RNN | MinSeq=5, MaxSeq=5, pre + postpartum, last HL | 512 | 0.390 ± 0.165 | 55.3 ± 11.5 |
| BCS CNN-RNN | MinSeq=1, MaxSeq=10, pre + postpartum, last HL | 512 | 0.374 ± 0.153 | 54.5 ± 10.3 |

**Table 4.10.** Results of SCK prediction models trained using depth image features and tabular data. DV stands for depth vectors, SR stands for sampling resolution, norm stands for normalized, DA stands for depth areas, and the CNN-RNN features reported were extracted from the last hidden state of the last LSTM layer of the model trained using only sequences of three prepartum frames. The $F_1$ scores and accuracies are reported as (mean ± standard deviation) across 20 random iterations of training and testing splits. The best results for each metric, as well as the models that achieved the highest $F_1$ score for SCK prediction with and without including DMI, are highlighted in bold. The model including normalized depth vectors with a sampling resolution of 20 achieved the best performance when also including DMI variables (average $F_1$ score = 0.706), and the model including PCA-transformed normalized depth vectors with a sampling resolution of 200 achieved the best performance when not including DMI variables (average $F_1$ score = 0.596). These models possibly benefited from including information about the cow body shape in a more direct way through depth values between anatomical keypoints, as opposed to the more indirect deep neural network features.

| Image features | PCA components (for image features only) | Include DMI? | Total number of features | $F_1$ score (mean ± SD) | Accuracy % (mean ± SD) |
|---|---|---|---|---|---|
| None | No PCA | Yes | 25 | 0.655 ± 0.094 | 74.2 ± 8.0 |
| CNN, CORN | 25 | Yes | 50 | 0.659 ± 0.091 | 73.2 ± 7.2 |
| **DV, SR=20 norm** | **No PCA** | **Yes** | **685** | **0.706 ± 0.125** | 76.8 ± 8.7 |
| DV, SR=200 norm | 25 | Yes | 50 | 0.680 ± 0.156 | **77.6 ± 8.1** |
| DA, SR=200 norm | No PCA | Yes | 58 | 0.643 ± 0.127 | 74.2 ± 8.1 |
| CNN-RNN, 3-3, pre, last HL | 25 | Yes | 50 | 0.649 ± 0.081 | 72.9 ± 6.9 |
| None | No PCA | No | 23 | 0.547 ± 0.140 | 67.4 ± 8.6 |
| CNN, CORN | 23 | No | 46 | 0.524 ± 0.163 | 67.4 ± 12.7 |
| DV, SR=20 norm | No PCA | No | | 0.486 ± 0.169 | 61.3 ± 11.3 |
| **DV, SR=200 norm** | **23** | **No** | **46** | **0.596 ± 0.146** | **72.6 ± 6.4** |
| DA, SR=200 norm | No PCA | No | 56 | 0.507 ± 0.133 | 67.4 ± 9.5 |
| CNN-RNN, 3-3, pre, last HL | 23 | No | 46 | 0.475 ± 0.156 | 65.3 ± 10.4 |

**Table 4.11.** Results of SCK prediction models trained using text embeddings and tabular data. The text embeddings are described in more detail in Table 4.3 and the *Text Feature Extraction* section. Models using *template text* or *combined text* were trained and validated using 5 samples per cow because 5 different templates were utilized for generating text from tabular data. The $F_1$ scores and accuracies are reported as (mean ± standard deviation) across 20 random iterations of training and testing splits. The best results for each metric, as well as the models that achieved the highest $F_1$ score for SCK prediction with and without including DMI, are highlighted in bold. The model using PCA-transformed notes text embeddings concatenated with tabular data achieved the best performance (average $F_1$ score = 0.681), surpassing the model trained using only tabular data. This indicates that the notes recorded in the farm management software contain important information for SCK prediction, and LLMs provide a way to include this information in quantitative analyses such as the machine learning pipeline proposed in this study.

| Text embeddings | PCA components (for text embeddings) | Include tabular data? | Total number of features | Samples per cow | $F_1$ score (mean ± SD) | Accuracy % (mean ± SD) |
|---|---|---|---|---|---|---|
| None | No PCA | Yes | 25 | 1 | 0.655 ± 0.094 | 74.2 ± 8.0 |
| Template text | No PCA | No | 1,536 | 5 | 0.593 ± 0.201 | 70.5 ± 11.8 |
| Template text | 25 | No | 25 | 5 | 0.600 ± 0.164 | 69.7 ± 11.4 |
| Notes text | No PCA | Yes | 1,561 | 1 | 0.621 ± 0.094 | 73.7 ± 8.0 |
| **Notes text** | **25** | **Yes** | **50** | **1** | **0.681 ± 0.209** | **78.7 ± 11.0** |
| Combined text | No PCA | No | 1,536 | 5 | 0.517 ± 0.114 | 62.6 ± 7.6 |
| Combined text | 25 | No | 25 | 5 | 0.518 ± 0.150 | 65.8 ± 11.9 |

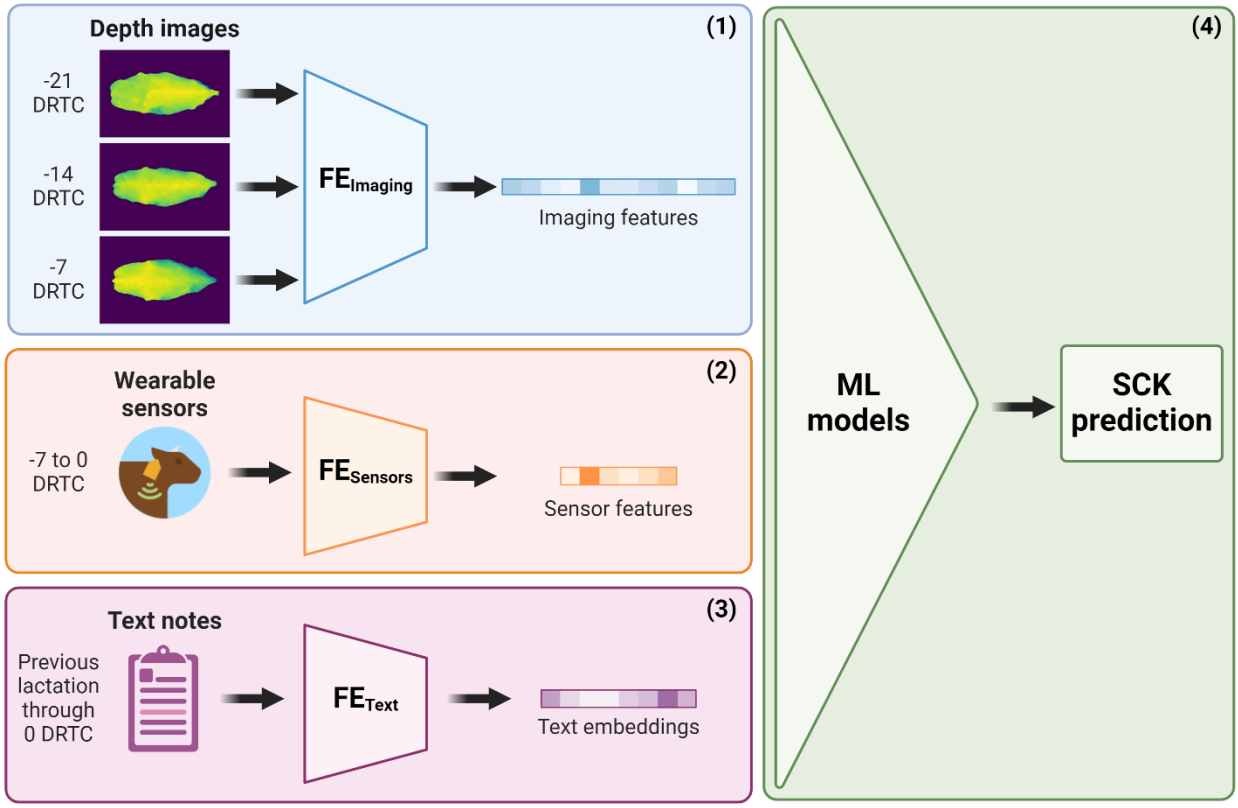



**Figure 4.1.** Overview of the first four steps of the proposed machine learning pipeline: (1) using deep learning and image processing techniques to extract features related to body shape from depth images collected from dairy cows during prepartum; (2) calculating descriptive features from prepartum feeding behavior, cow activity, and cow history data; (3) extracting features from textual data using LLMs; (4) integrating all the extracted features into machine learning models that predict, using exclusively prepartum data, the cows with a high risk of developing subclinical ketosis during the first 15 days of lactation. $FE_{Imaging}$, $FE_{Sensors}$, and $FE_{Text}$ represent the feature extractors utilized for depth images, cow behavior and history data, and textual data, respectively.

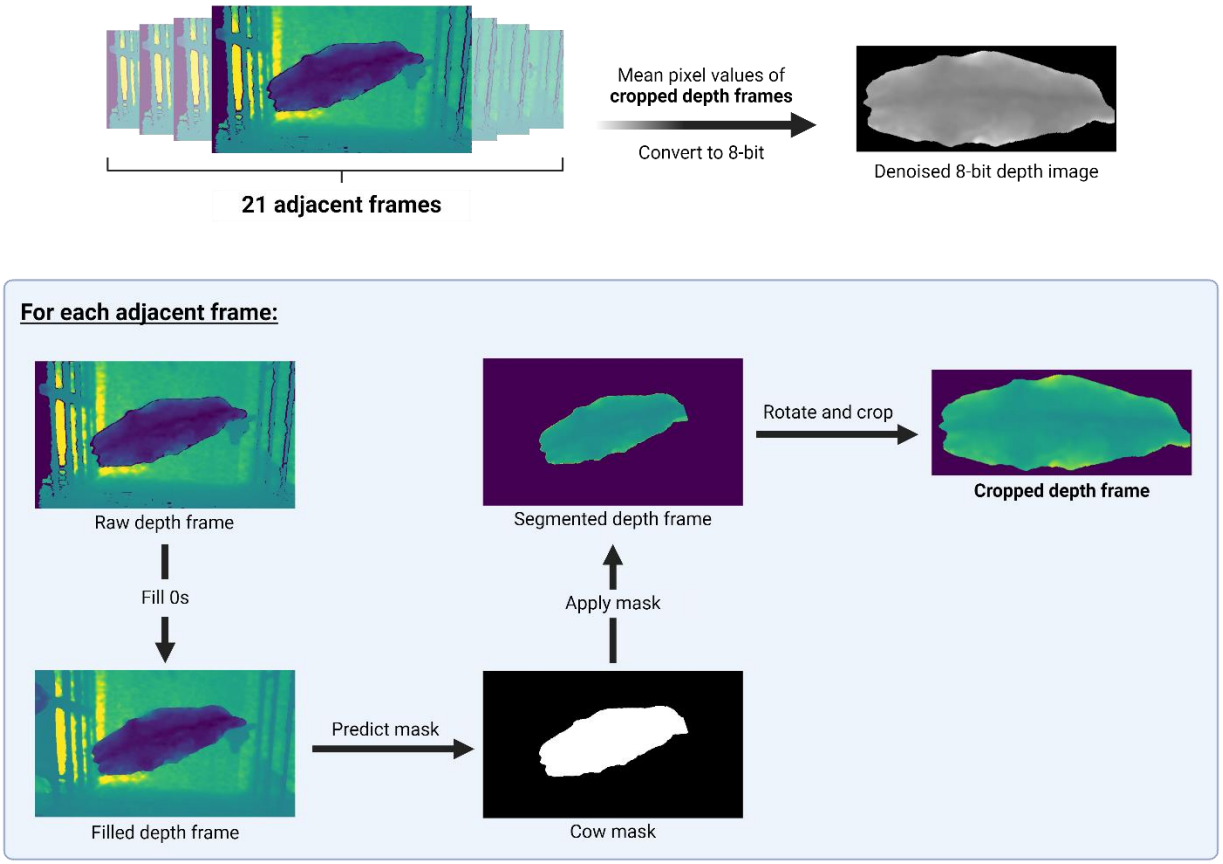
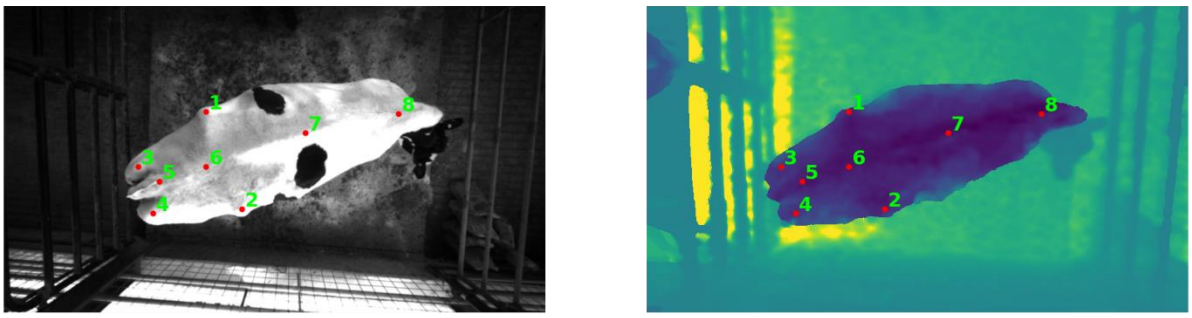



**Figure 4.2.** Image processing pipeline for generating rotated, cropped, and denoised 8-bit images containing the segmented body surface of a cow. The depth frames were denoised by using depth frames that were adjacent in the recorded video. Each adjacent depth frame was segmented using a trained U-net model for cow body segmentation, rotated and cropped around the cow body. The mean pixel values of the cropped adjacent depth frames were calculated to generate a final denoised depth image, which was then converted to an 8-bit image. The generated 8-bit images were used for training and testing the CNNs for BCS prediction, and for feature extraction.




(a) Infrared image containing the eight defined anatomical keypoints

(b) Depth image containing the eight defined anatomical keypoints

**Figure 4.3.** Infrared (a) and corresponding depth (b) images containing anatomical keypoints defined on the back of the cows. The defined keypoints included the (1) left and (2) right hooks, (3) left and (4) right pin bones, (5) tailhead, (6) sacral vertebrae, (7) lumbar vertebrae, and (8) cervical vertebrae. These keypoints were automatically detected for each depth frame using a trained keypoint detection YOLOv8 model. Multiple 1D depth vectors were calculated by sampling depth values between keypoint pairs, which served as image features for subclinical ketosis detection.

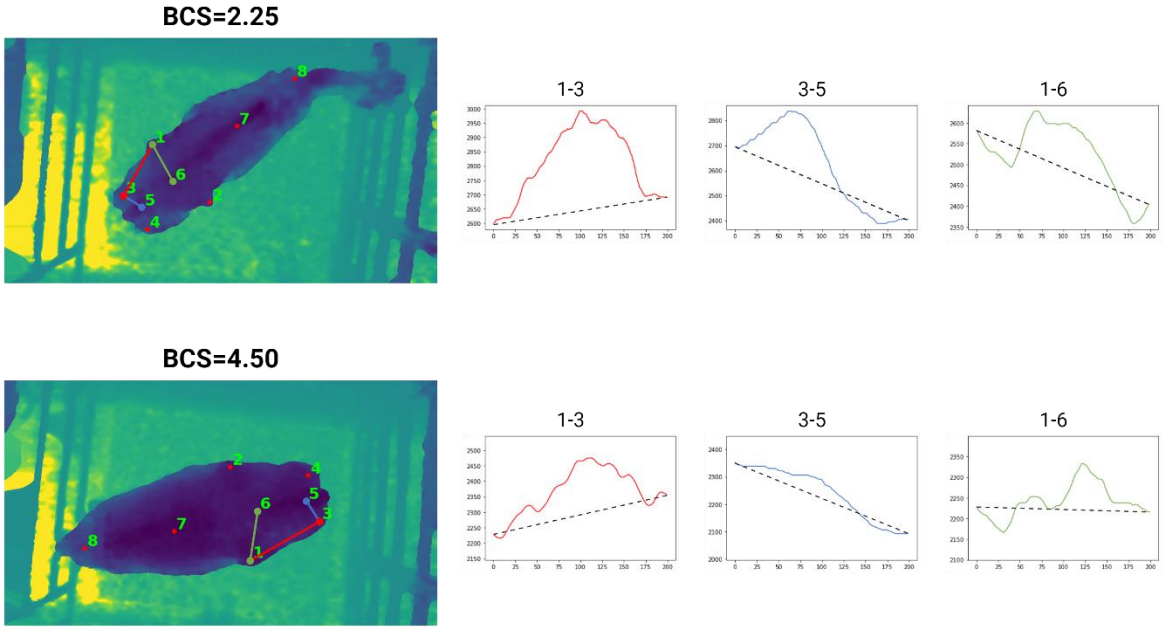



**Figure 4.4.** Examples of 1D depth vectors extracted from different frames by sampling the depth values between pairs of keypoints. The shapes of those depth vectors varied considerably between a cow with body condition score 2.25 (top image) and a cow with body condition score 4.50 (bottom image). The dashed lines connect the first and last sampled depth values from each pair, which were used to normalize the depth vectors and to calculate the areas under these vectors, which were also used as features. This illustration was constructed using a sampling resolution of 200.

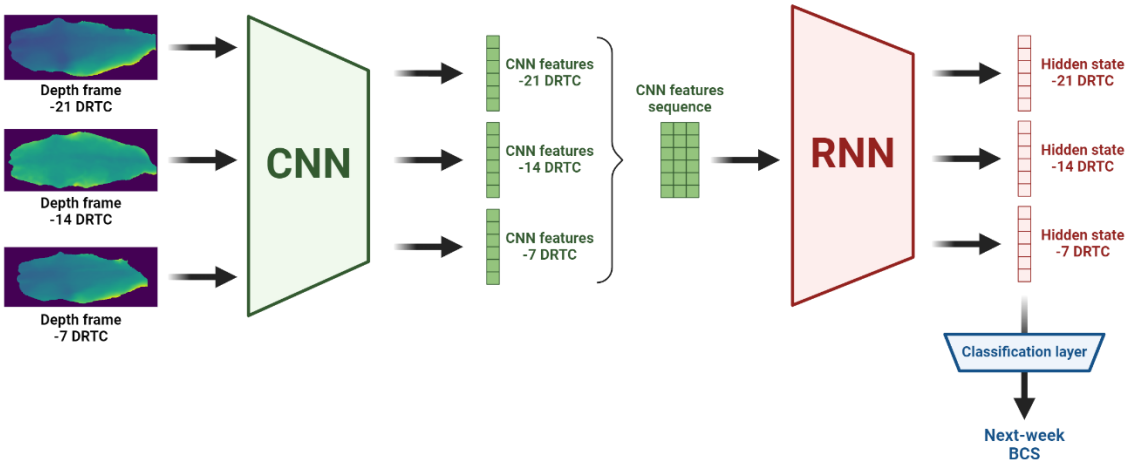



**Figure 4.5.** Overview of the CNN-RNN architecture and feature extraction methods. Depth frames from the same cow on consecutive weeks are passed to a CNN that extracts features from each of them. The CNN features are then passed to the RNN as a sequence, which outputs the hidden states of each frame in the sequence. The last (time-wise) hidden state is passed to a classification layer that outputs the prediction for the BCS of that cow on the following week in relation to the last date of the sequence. Using this CNN-RNN approach, two different ways to extract features from a sequence of depth frames were explored: concatenating the hidden states from all images in the sequence or retrieving just the last hidden state output by the RNN.

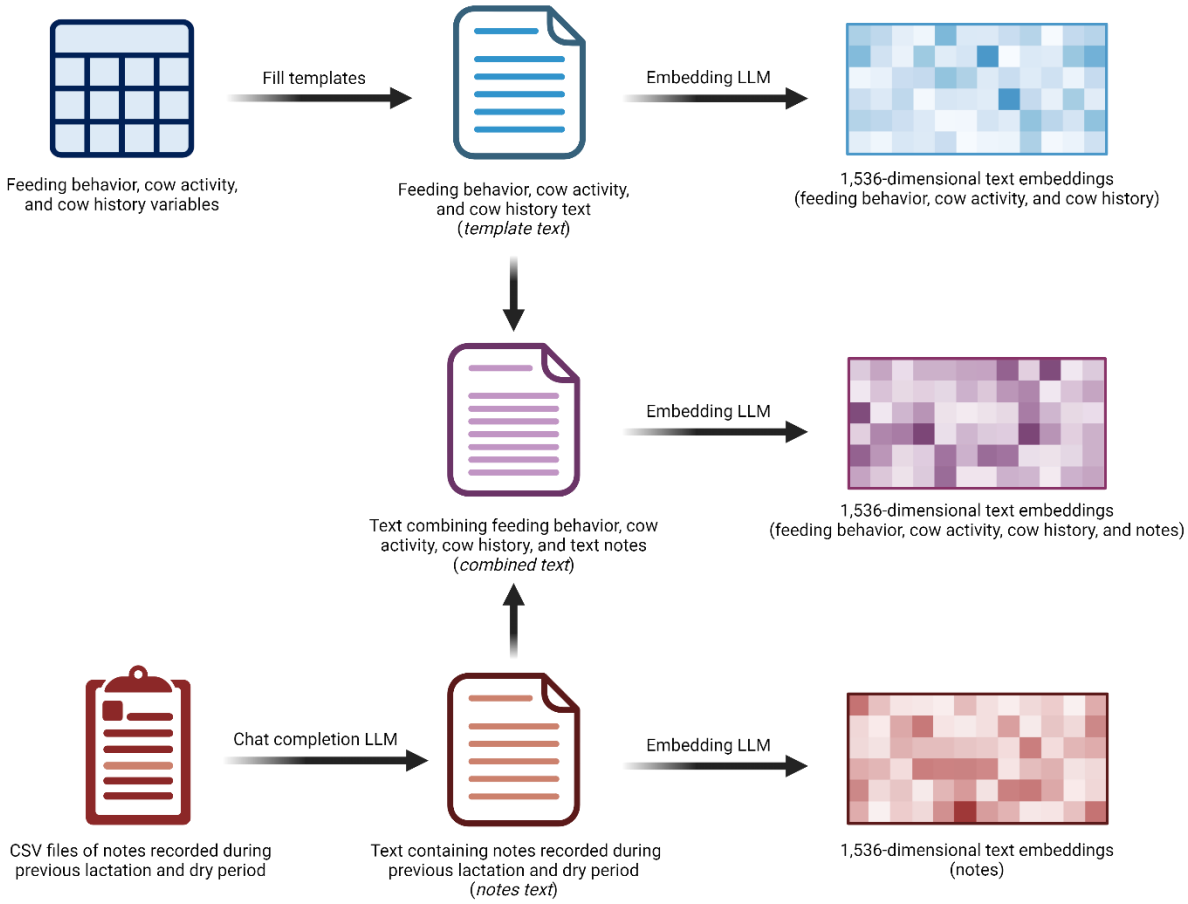



**Figure 4.6.** Procedures to extract the three feature sets from textual data. Text embeddings were extracted from text generated by inserting tabular data into templates (*template text*); text generated from notes retrieved from the farm management software (*notes text*); and a combination of these two texts (*combined text*). When training the SCK prediction model, text embeddings extracted from notes text were concatenated with the tabular data containing behavior and historical information.

| DATE | EVENT | REMARKS | DETAILS | RESPONSIBLE[1] | DIM | PEN |
|---|---|---|---|---|---|---|
| 9/14/2019 | FRESH | 9762/92 | Heifer 9762 Live | | - | 26 |
| 9/17/2019 | LAME | EXD1.21 | FOOT ROT EXED | | 3 | 9 |
| 12/5/2019 | BRED | 511H12240 | Open (O), Double Ovsynch (D) | Rafael | 82 | 34 |
| 1/6/2020 | RECHK | LOSING? | - | | 114 | 34 |
| 1/13/2020 | OPEN | LUT2CLEA | - | | 121 | 34 |
| 1/20/2020 | NOTES | CIDR | - | | 128 | 34 |
| 1/27/2020 | OK | LUT | - | | 135 | 34 |
| 1/30/2020 | BRED | 629H18813 | Open (O), LUT (L) | Joao | 138 | 34 |
| 3/2/2020 | OPEN | LUT | - | | 170 | 34 |
| 3/5/2020 | BRED | 11AN1212 | Open (O), Resynch 25 (T) | Rafael | 173 | 34 |
| 4/6/2020 | OPEN | CL RIGHT | - | | 205 | 34 |
| 4/9/2020 | BRED | 29AN1993 | Open (O), Resynch 25 (T) | Joao | 208 | 34 |
| 5/11/2020 | OPEN | LUT | - | | 240 | 34 |
| 5/14/2020 | BRED | 829AN1868 | Pregnant (P), Resynch 25 (T) | Joao | 243 | 34 |
| 6/15/2020 | PREG | ROCL | - | | 275 | 34 |
| 6/29/2020 | PREG | ROCL | - | | 289 | 34 |
| 7/1/2020 | MOVE | TOMARS | - | | 291 | 34 |
| 7/17/2020 | LAME | LATDRH | Dig Derm - Wa | | 307 | 94 |
| 7/20/2020 | PREG | HEIFER | - | | 310 | 94 |
| 9/14/2020 | PREG | 123 Days | - | | 366 | 94 |
| 12/27/2020 | MOVE | F094T168 | 94 → 168 | | 470 | 94 |
| 12/30/2020 | DRY | SPCDC | SPECTRA-DC.IM | | 473 | 168 |
| 1/6/2021 | MOVE | TO ARL | - | | 480 | 168 |
| 1/26/2021 | MOVE | CLOSEUP | - | | 500 | 38 |
| 1/26/2021 | ONEXP | HMW618 | - | | 500 | 26 |
| 2/2/2021 | NOTES | URINE 5.5 | - | | 507 | 26 |

(a) CSV file containing all the notes taken during the previous lactation

The chronological report of events for the cow described in the CSV is as follows:

1. **September 14, 2019**: The cow entered the herd as a fresh heifer (9762/92). This event happened in pen 26.
2. **September 17, 2019**: The cow was diagnosed as lame due to foot rot (EXD1.21) on day 3 of lactation in pen 9.
3. **December 5, 2019**: The cow was bred by Rafael using semen from bull 511H12240 with the method "Open (O), Double Ovsynch (D)" on day 82 of lactation in pen 34.
4. **January 6, 2020**: A recheck (RECHK) indicated a possible issue ("LOSING?") on day 114 in pen 34.
5. **January 13, 2020**: The cow was confirmed open with a treatment of LUT2CLEA on day 121 in pen 34.
6. **January 20, 2020**: Notes indicate a CIDR intervention on day 128 in pen 34.
7. **January 27, 2020**: The cow was OK'd following LUT treatment on day 135 in pen 34.
8. **January 30, 2020**: The cow was bred by Joao using semen from bull 629H18813 with the method "Open (O), LUT (L)" on day 138 in pen 34.
9. **March 2, 2020**: The cow was confirmed open following LUT treatment on day 170 in pen 34.
10. **March 5, 2020**: The cow was bred by Rafael using semen from bull 11AN1212 with the method "Open (O), Resynch 25 (T)" on day 173 in pen 34.
11. **April 6, 2020**: The cow was confirmed open with a note of "CL RIGHT" on day 205 in pen 34.
12. **April 9, 2020**: The cow was bred by Joao using semen from bull 29AN1993 with the method "Open (O), Resynch 25 (T)" on day 208 in pen 34.
13. **May 11, 2020**: The cow was confirmed open following LUT treatment on day 240 in pen 34.
14. **May 14, 2020**: The cow was bred by Joao using semen from bull 829AN1868 with the method "Pregnant (P), Resynch 25 (T)" on day 243 in pen 34.
15. **June 15, 2020**: Pregnancy confirmed (ROCL) on day 275 in pen 34.
16. **June 29, 2020**: Another pregnancy check confirmed (ROCL) on day 289 in pen 34.
17. **July 1, 2020**: The cow was moved to Mars on day 291 in pen 34.
18. **July 17, 2020**: The cow was diagnosed as lame due to digital dermatitis (Dig Derm - Wa) on day 307 in pen 94.
19. **July 20, 2020**: Pregnancy confirmed with a note of "HEIFER" on day 310 in pen 94.
20. **September 14, 2020**: Pregnancy progression noted as 123 days on day 366 in pen 94.
21. **December 27, 2020**: The cow was moved from pen 94 to 168 on day 470.
22. **December 30, 2020**: The cow was dried off with SPECTRA-DC.IM on day 473 in pen 168.
23. **January 6, 2021**: The cow was moved to ARL on day 480 in pen 168.
24. **January 26, 2021**: The cow was moved to the close-up pen on day 500 in pen 38 and also noted for on-experiment with HMW618 in pen 26.
25. **February 2, 2021**: Notes indicate a urine pH of 5.5 on day 507 in pen 26.

This report tracks the cow's journey through various health checks, breeding attempts, pregnancy confirmations, and management changes throughout her lactation period.

The cow's current lactation, for which we want to predict the risk of subclinical ketosis, started on February 13, 2021.

(b) Free text generated from the CSV file using OpenAI's chat completion API

**Figure 4.7.** Example of (a) a CSV file containing notes taken during a cow's previous lactation and dry period and (b) the corresponding text generated using OpenAI's chat completion API. The texts were generated using the *GPT-4* model, a temperature of 0.5, and the following system and user prompts: *‘"DIM" means the number of days in lactation that the cow had when that event happened. "PEN" is the pen number where that event occurred.’* and *‘Give me a chronological report of events that happened to the cow described in this CSV: "{CSV content}"’*. [1]The names in the RESPONSIBLE column were replaced with the authors' names to keep the privacy of the corresponding farm employees.

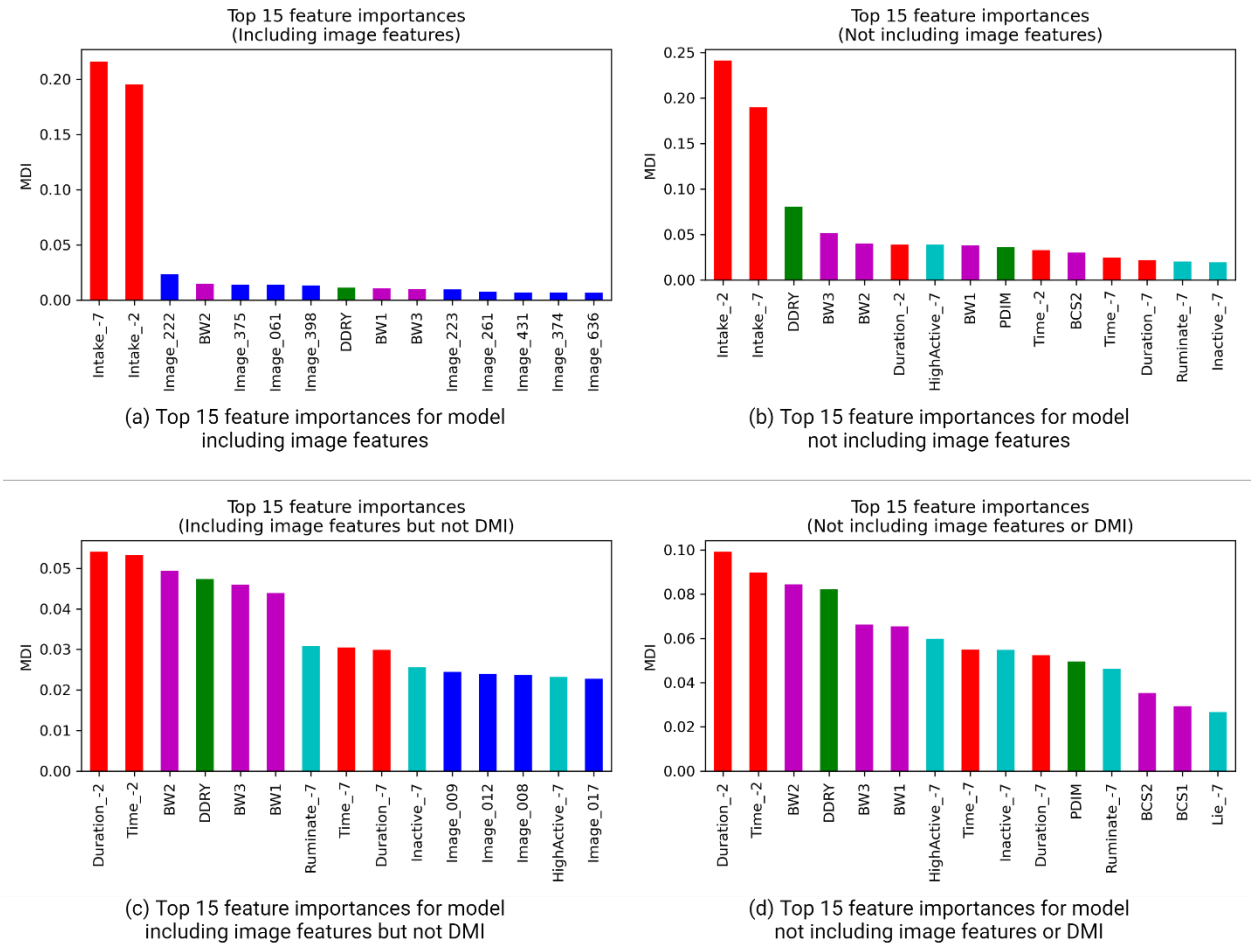



**Figure 4.8.** Average feature importances of the random forest SCK prediction models (a and c) including and (b and d) not including depth image features, and (a and b) including and (c and d) not including DMI. DMI measurements prove important for SCK prediction, followed by the previous days dry (*DDRY*) and the three body weight measurements (*BW1*, *BW2*, and *BW3*), which feature among the top 15 features both when including or not depth image features. Nine and four image features were included among the top 15 features when including or not DMI, respectively, which, along with the increase in average $F_1$ score, further highlights the importance of including depth image analysis in the machine learning pipeline for SCK prediction. Feature importances were calculated as the mean decrease in impurity of each feature using the Gini criterion.

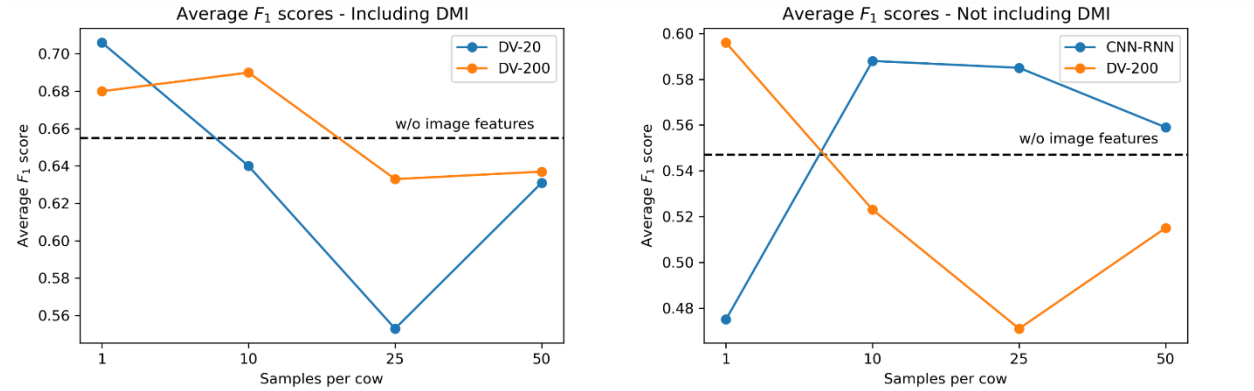



**Figure 4.9.** Average $F_1$ scores for each number of samples per cow, (a) including or (b) not including DMI. Only the best two models for each analysis (including or not DMI) were plotted. When using 1 sample per cow, the mean feature values were calculated based on 50 random variations of each cow. The DV-20 and DV-200 models were trained using normalized depth vectors with 20 and 200 sampling resolution respectively. The CNN-RNN models were trained using the same CNN-RNN features as in the second comparative analysis (*Combining Image Features with Tabular Data*). Dashed lines represent the performance of the baseline models containing only tabular data. In general, including more samples per cow hindered the performance of the models, except for when using CNN-RNN features.

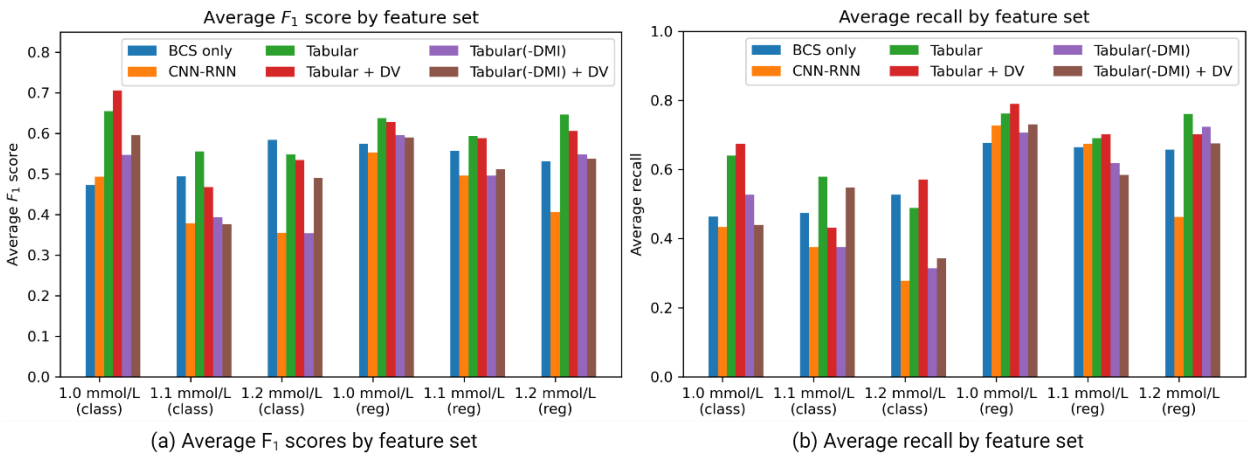



**Figure 4.10.** Average (a) $F_1$ and (b) recall scores for different feature sets, plasma BHB thresholds, and training objectives (classification or regression). *Class* stands for classification and *reg* stands for regression; the *Tabular* variables correspond to tabular data, DV stands for depth vector, and feature sets containing (-DMI) did not include the two DMI variables. The CNN-RNN feature set contained features extracted from the last hidden state of the CNN-RNN next-week BCS prediction using only sequences containing three prepartum images. When including DMI, the depth vectors were normalized depth values with sampling resolution of 20 without PCA, and when not including DMI the depth vectors were normalized depth values with sampling resolution of 200 applying PCA with 23 components. The best performing model based on $F_1$ score was achieved using Tabular+DV features for binary classification using a BHB threshold of 1.0 mmol/L (average $F_1$ score = 0.706). Including image features resulted in a decrease in the $F_1$ score of all models except for when performing a regression and a threshold of 1.1 mmol/L without including DMI, when performing classification with a threshold of 1.2 mmol/L without including DMI, and when performing classification with a threshold of 1.0 mmol/L in all cases. However, when looking at recall, including image features resulted in a higher score in most cases, including the highest recall achieved when using Tabular+DV features for regression and a threshold of 1.0 mmol/L (average recall = 0.790).

# CHAPTER FIVE: CLOUD COMPUTING FRAMEWORK FOR AUTOMATED PHENOTYPE COLLECTION, INTEGRATION, AND DATA ANALYSIS IN DAIRY SYSTEMS


## ABSTRACT

In precision livestock farming (**PLF**), wearable sensors, computer vision, and genomic tests generate large amounts of data that can be challenging to integrate and analyze jointly due to their diverse natures. At the same time, incorporating genomic and phenomic data together can be beneficial for developing predictive models in animal biology. The development of automated and modularized data pipelines using scalable solutions such as cloud computing can be an effective strategy to integrate and analyze animal-level information in real-time. The objectives of this study were (1) to propose a cloud computing-based framework to automate the processing and integration of phenotypic and genotypic data, and (2) to assess different data fusion strategies (early and late fusion, and cooperative learning) for the early detection of subclinical ketosis (**SCK**) in dairy cows, integrating wearable sensors, imaging systems, and genotypic data in livestock farms. We developed a modularized pipeline for image analyses including: body segmentation, frame quality assessment, animal identification, and body condition score (**BCS**), which were crucial to produce the features used for SCK detection. The body segmentation module achieved a Dice similarity coefficient of 0.990, the frame quality assessment module achieved an accuracy of 99.1%, the animal identification module achieved an accuracy of 93.2%, and the BCS module achieved accuracies of 81.1% and 96.2% when allowing up to 0.25 and 0.50 prediction error. For SCK detection, early fusion and cooperative learning achieved the lowest mean absolute errors on the prediction of plasma beta-hydroxybutyrate as a continuous variable (down to 0.242), and late fusion coupled with an ordinary least squares regression achieved the highest $F_1$ scores

for SCK binary prediction (up to 0.750). These results indicate that data fusion techniques can be used for efficient integration of genotypic and phenotypic data from multiple sensors. Additionally, SCK detection can be performed in dairy farms via the proposed cloud computing-based framework implemented using modular independent services, which can be customized and re-used for a variety of tasks.

## INTRODUCTION

As the global human population grows, so does the demand for food, putting pressure on food production systems to become more cost-effective and efficient, both in terms of production and environmental impact. In livestock systems, achieving such efficiency primarily hinges on optimized management practices and genetic improvement. Both strategies benefit significantly from high-throughput phenotyping, which involves the measurement and monitoring of key traits in living organisms in a way that is non-invasive, automated, and scalable (Koltes et al., 2019). Specifically, through precision livestock farming (**PLF**) technologies, high-throughput phenotyping enables more informed and rapid farm management decisions (Berckmans, 2017) and enhances the capabilities for genetic selection (Brito et al., 2020; Silva et al., 2021).

Precision livestock technologies provide a great way to implement high-throughput phenotyping, with computer vision-based systems emerging as potential approaches that provide non-invasive, automated, and scalable solutions for individual animal monitoring (Fernandes et al., 2020). However, a great challenge in applying PLF technologies in livestock farms pertains to making efficient use of the generated data (Koltes et al., 2019). With data generated by each PLF system typically being available locally at the farm, locked within each provider's software, and/or following each company's proprietary formats, data availability and integration become big

challenges in the adoption and development of new solutions that could arise from analyzing data from multiple sources simultaneously (Neethirajan and Kemp, 2021).

Cloud computing technology could be used to store the data generated by PLF systems available on the farm and process such data into valuable information for the farmer, which could be accessed from anywhere with internet connection. Integrating PLF technologies into cloud computing solutions can mitigate some of the problems related to data integration and availability, as all data generated from different sources are stored and made available on a single platform (Schokker et al., 2022). Edge computing is also considered a powerful alternative for processing and distributing data collected at farms (Alonso et al., 2020), but it lacks the flexibility that cloud computing provides to scale up computing power for data analytics, predictive modeling, and other data processing requirements on demand.

Although cloud computing provides great infrastructure for PLF systems, integrating multiple data modalities into predictive analyses can still pose challenges due to the diversity of data structures and representations in data originating from heterogeneous sources (Atrey et al., 2010). For example, data produced by imaging systems and wearable sensors might have completely different data acquisition frequencies, structural patterns, and, ultimately, ways to process and extract information from. In agriculture, genomic and phenomic data contain distinct signals that can be combined to train more robust models than if these modalities were used independently. The integration of these data modalities can be performed either via early fusion, which consists of concatenating the features from different modalities into a single joint representation for prediction, late fusion, which consists of integrating the predictions made for each modality separately, or hybrid fusion, which combines the concepts of both early and late fusion approaches (Baltrušaitis et al., 2019). Recently, Ding et al. (2022) proposed a hybrid fusion

method called cooperative learning, which merges multiple modalities in a data-adaptive manner, introducing an agreement penalty that encourages predictions from separate modalities to reach a consensus. The weight of this agreement penalty in relation to a regularized squared-error loss dictates how much the final model relies on early and late fusion. This weight is considered a hyperparameter during model training, and it is adjusted based on a validation set or cross-validation. Evaluating different data fusion techniques is critical for advancing omics integration in agriculture, given the variety of data modalities obtained through sensing technologies. Moreover, the integration of such techniques with cloud computing technology provides a powerful tool for the deployment of automated PLF systems that optimize livestock management decisions and improve animal health and welfare. An overview of the fusion techniques explored in this study is illustrated in Figure 5.1.

Integrating genotypic and phenotypic data in the proposed modularized cloud computing framework allows for the development of a variety of predictive algorithms for improving animal monitoring and farm management practices. We propose a framework for extracting features from different modalities (genotypic data, imaging data, and phenotype tabular data) and combining these features for phenotype prediction using data fusion and machine learning techniques. This framework can be reused and expanded to solve different problems in livestock farms that are related to body shape, genetics, and animal behavior. In this study, we have explored integrating genotypic data, wearable sensors, and imaging systems for the early detection of subclinical ketosis (**SCK**), which is one of the most prevalent and economically detrimental peripartum disorders affecting dairy cows (Cainzos et al., 2022). Many of the image processing modules implemented for the early detection of SCK were also used in the cloud computing framework for performing

automatic individual animal identification and body condition score (**BCS**) evaluation, demonstrating the reusability of the modules implemented in the proposed framework.

The objectives of this study were (1) to propose a cloud computing-based framework to automate the processing and integration of phenotypic and genotypic data, and (2) to assess different data fusion strategies (early and late fusion, and cooperative learning) for the early detection of SCK in dairy cows, integrating wearable sensors, imaging systems, and genotypic data in livestock farms. We demonstrate the proposed framework by implementing automated animal identification and BCS assessment through imaging data, and a novel approach for the early detection of SCK in dairy cows.

## MATERIAL AND METHODS

The data processing and phenotype prediction pipeline developed in this study consisted of four steps: (1) feature extraction from genotype data; (2) feature extraction and prediction from image data; (3) feature engineering from sensor and management software data; and (4) data fusion and machine learning model prediction. The image processing pipeline was further divided into four separate procedures: (1) cow body segmentation; (2) image quality classification; (3) animal identification; and (4) BCS classification. It is worth noting that, although the main phenotype explored in this work is the early detection of subclinical ketosis, the same data processing procedures can be applied to any phenotype that is correlated to genetic data, body shape changes, and animal behavior, which makes the developed pipeline reusable for other applications. In addition, animal identification is an essential step for any individual phenotyping tool that utilizes images. Here we perform animal identification directly from images, without the need for an external animal identification system such as radio frequency identification devices.

### BCS Assessment and Subclinical Ketosis Classification

Three trained independent evaluators determined the BCS of 115 multiparous cows weekly from 21 to 7 days before the expected calving date, and from 7 to 56 days after calving using a 5-point scale (Wildman et al., 1982). The BCS in quarter-point increments that was closest to the average among the three evaluators was then determined for each cow and date. The assessed BCS values were 2.00 (n = 1), 2.25 (n = 18), 2.50 (n = 42), 2.75 (n = 123), 3.00 (n = 222), 3.25 (n = 247), 3.50 (n = 218), 3.75 (n = 139), 4.00 (n = 84), 4.25 (n = 43), and 4.50 (n = 27).

Blood samples from 106 of those cows were collected at 3, 5, 7, 11, and 14 days after the calving date. Concentrations of plasma beta-hydroxybutyrate (**BHB**) were quantified using the Catachem ChemWell-T analyzer (Catachem; Oxford, CT) as previously described (Holdorf et al., 2023). Cows with the maximum measured BHB concentration among those five samples above 1.0 mmol/L were classified as having a postpartum subclinical ketosis event. It is worth noting that commonly used BHB thresholds for ketosis detection range between 1.0 and 1.2 mmol/L. However, in our case, using thresholds higher than 1.0 mmol/L resulted in unbalanced datasets, with only a few cases of cows exceeding such thresholds and being considered an SCK event.

Genotype Data

Genotypic information from 163 cows was sourced from the Council on Dairy Cattle Breeding (**CDCB**) database, resulting in 78,964 single-nucleotide polymorphism (**SNP**) markers per cow. Of those 163 animals, 19 were separated as testing cows, and the remaining 144 were used as a training set for further quality control analyses and feature extraction. SNPs with call rates below 0.95, minor allele frequencies below 0.01, or with a highly significant deviation from the Hardy-Weinberg equilibrium (p-value below $10^{-5}$) were removed from further analyses, resulting in 73,031 SNPs. Missing SNP values were imputed with the predominant value for the corresponding marker in the training dataset, and each SNP value was converted to two binary

values, with the homozygous set to *00* and *11*, and the heterozygous set to *01*. The resulting 146,062 binary values for each of the 144 cows were used to train a Uniform Manifold Approximation and Projection (**UMAP**) (McInnes et al., 2018) model with local neighborhood size of 15 samples, cosine similarity as the distance metric, and 128 components as the output dimensionality. UMAP is a commonly used technique for reducing data dimensionality in a nonlinear manner in the context of cellular biology and genomics, often surpassing the performance of other dimensionality reduction tools for clustering, visualization, and classification (Becht et al., 2019; Allaoui et al., 2020; ElKarami et al., 2022). The trained UMAP model was used to extract 128 features from both the 144 training cows and the 19 testing cows.

## Image Processing

### Cow Body Segmentation

Videos of 74 pre-weaned Holstein dairy calves, aged two to eight weeks and housed at the Emmons Blaine Dairy Cattle Research Center (Arlington, WI), were captured from a top-down view while individually weighing each animal. Among these, 43 calves were recorded using a Kinect V2 sensor (Microsoft; Redmond, WA) at a resolution of 512 × 424 pixels, while the remaining 31 were recorded using Intel RealSense D435 depth-sensing cameras (Keselman et al., 2017) at a resolution of 848 × 480 pixels. Using the same Intel RealSense cameras at the same resolution, videos from 155 multiparous Holstein cows were captured as they entered the milking parlor, or while individually weighing each animal. Additionally, snapshots from 59 of those cows were automatically captured at a resolution of 640 × 480 as they walked under Intel RealSense D435 cameras installed at the milking parlor exit lanes.

All of those videos and snapshots contained a depth channel with each pixel consisting of a 16-bit unsigned integer value representing the distance in millimeters between the object at that

pixel and the camera lenses. Depth frames were extracted from the videos, and each frame was then clipped to floating-point values between 0 and 1 by dividing each pixel value by 5,000 for the cow videos taken while weighing (the camera was positioned slightly above 5 meters from the scale flooring), or by normalizing each pixel value using the corresponding $1^{st}$ and $99^{th}$ percentiles for all remaining videos or snapshots. Each frame was then converted to an 8-bit grayscale image by multiplying their pixel values by 255 and converting the results to 8-bit unsigned integers.

Segmentation masks from random depth frames were manually annotated containing the full animal body excluding the neck and the head. Care was taken to obtain a balanced number of frames per animal and date, as some animals had images collected on multiple different days, resulting in 968 images of calves, 159 images of cows entering the milking parlor, 248 images of cows at the milking parlor exit lanes, and 2,328 images of cows at the weighing scale. From those images, two distinct testing sets were established: 124 randomly selected images from the milking parlor exit lanes (*test_seg_lane*); and 462 images of 21 predetermined testing cows at the scale, with 19 of them corresponding to the same testing cows determined for the genotype data (*test_seg_scale*). The remaining 3,117 images were used to train a deep neural network for cattle body segmentation based on the U-net architecture (Ronneberger et al., 2015), with training and validation sets defined through a random split of 90% of the images for training and 10% for validation.

The U-net model was trained for 100 epochs with batch size equal to 1 and an initial learning rate of $10^{-5}$. The model parameters were optimized for both minimizing the pixel level cross-entropy loss and maximizing the Dice similarity coefficient (Zou et al., 2004) using RMSProp (Hinton et al., 2012) with weight decay and momentum of $10^{-8}$ and 0.999, respectively. A learning rate scheduler was set to reduce the learning rate by a factor of 10 when the Dice

similarity coefficient calculated on the validation set would not increase for 4 consecutive epochs. Before each forward pass during training, the images were randomly flipped horizontally and vertically with 50% probability and rotated by a random angle between -90 and 90 degrees. The Dice similarity coefficient was used as the main metric to evaluate model performance on the training, validation, and both testing sets.

Image Quality Classification

Depth images from cows exiting the milking parlor were automatically collected as they walked through the four exit lanes using Intel RealSense D435 cameras at a resolution of 640 × 480 pixels, resulting in 20,170 images. Each depth image was manually annotated as being *good* or *bad* for further analysis, based on whether it contained the whole body of a single cow, which resulted in 12,368 *bad* and 7,802 *good* images. The depth images were normalized to pixel values between 0 and 1 using the corresponding $1^{st}$ and $99^{th}$ percentiles and converted to 8-bit grayscale images by multiplying their pixel values by 255 and converting the results to 8-bit unsigned integers. They were then segmented using the trained cow body segmentation model, rotated so that the major axis of the ellipse that had the same second-moments of the mask was parallel to the x-axis, and cropped around the bounding box containing all mask pixels.

The same depth frames that were collected from cows during individual weighing and manually segmented to train and test the cow body segmentation network underwent the following processing steps: pixel values were divided by 5,000 and clipped to values between 0 and 1, as the camera was positioned slightly above 5 meters from the scale flooring; the image was converted to 8-bit grayscale by multiplying the pixel values by 255 and converting the results to 8-bit unsigned integers; the corresponding manually annotated segmentation mask was applied to the image; the image was rotated so that the major axis of the ellipse that had the same second-

moments was parallel to the x-axis; and the image was cropped around the bounding box containing all the masked pixels. All 2,328 depth images obtained this way were considered *good*, so *bad* pairs were artificially crafted by cropping the area to the right of the line that connects one random pixel on the top row to another at the bottom row of the image, simulating situations where the cow body would be partially occluded. This resulted in 2,328 *good* and 2,328 *bad* images collected on the scale.

From those 24,826 annotated depth images, two distinct testing sets were constructed: all images collected from one of the four milking parlor exit lanes (4,976 images; *test_quality_lane*); and 462 images of the same 21 predetermined testing cows collected on the scale, along with their corresponding *bad* pairs (924 images; *test_quality_scale*). The remaining 18,926 images were used to train a deep neural network for image quality classification based on the ResNet-50 architecture (He et al., 2016), with training and validation sets defined through a random split of 80% of the images for training and 20% for validation.

The image quality classification model was constructed using a ResNet-50 network (He et al., 2016) pretrained on the ImageNet dataset (Deng et al., 2009), and the output layer was replaced by a fully-connected layer with two output neurons. The training process was performed via a two-stage approach: feature extraction and fine-tuning. In the feature extraction stage, the network was trained for 30 epochs keeping the weights from all except the output layer frozen, allowing features previously learned through ImageNet to be used and retained. In the fine-tuning stage, weights from all layers were unfrozen and the network was trained for 60 epochs with a smaller learning rate, allowing it to learn features that are more specific to the current task. The batch size in both stages was set to 16, and the initial learning rates for feature extraction and fine-tuning were $10^{-3}$ and $10^{-4}$, respectively. The model parameters were optimized for minimizing the cross-entropy

loss using Adam (Kingma and Ba, 2014) with running average coefficients of 0.9 and 0.999, and a scheduler was set to reduce the learning rate by a factor of 10 every 6 epochs. Before each forward pass, the images were resized to 224 × 224 keeping the original aspect ratio by padding the smallest dimension, and randomly flipped horizontally and vertically with 50% probability. Model accuracy was monitored for both training and validation sets during training, and the final model accuracy was calculated for both independent testing sets.

Animal Identification

Videos of 90 multiparous Holstein cows were manually collected during weekly individual weighing from 21 to 7 days before the expected calving date and from 7 to 56 days after calving using an Intel RealSense D435 depth-sensing camera, resulting in 11 videos per cow. These videos were also used for cow body segmentation and image quality classification, described in the two previous subsections (*Cow Body Segmentation* and *Image Quality Classification*). At the time of recording, the videos were labeled with the corresponding cow tag identification number. From each video, 10 random infrared frames were extracted, each represented by an 8-bit grayscale image captured by one of the Intel RealSense infrared sensors, resulting in 9,680 infrared frames because 22 videos did not contain an infrared channel and were excluded from the animal identification analysis. The corresponding depth frames were passed through the cow body segmentation model and the predicted segmentation masks were applied to the original infrared frames. The segmented infrared frames were then rotated so that the major axis of the ellipse that had the same second-moments of the mask was parallel to the x-axis, and cropped around the bounding box containing all mask pixels. This resulted in 9,679 successfully segmented infrared frames.

For training the animal identification neural network, frames from the first six videos of each cow were allocated to the training set (5,400 images), frames from the 7$^{th}$ video were allocated to the validation set (900 images), and the testing set was composed of frames from the last four videos of each cow (3,379 images), with the videos being recorded at one-week intervals. Similar to the image quality classification model, the animal identification model was constructed based on a pretrained ResNet-50 network (He et al., 2016), but with the output layer being replaced by a fully-connected layer with 90 output neurons instead (one for each cow). The training procedure and hyperparameters were similar to those used in the image quality classification model, with the distinction that the images were resized by stretching the originally smaller dimension instead of padding it, and an additional image augmentation step was performed to randomly jitter brightness, contrast, and saturation by 40% each.

BCS Classification

Videos of 115 multiparous Holstein cows were manually collected during weekly individual weighing from 21 to 7 days before the expected calving date and from 7 to 56 days after calving using an Intel RealSense D435 depth-sensing camera. Some animals did not have all 11 weekly videos recorded because they either calved more than a week before the expected calving date or were removed from the experiment before reaching 56 days after calving, resulting in a total of 1,164 videos. For each video, a procedure similar to those described in the two previous subsections (*Image Quality Classification* and *Animal Identification*) was performed: 10 random depth frames were extracted and normalized to an 8-bit image, the segmentation masks predicted using the cow body segmentation model were applied, and the depth images were rotated and cropped accordingly. This resulted in a total of 11,639 segmented depth frames.

For training the BCS classifier, frames from the same 21 testing cows as determined previously were separated to form a testing set, and the remaining frames were randomly split into training and validation sets in a 90 to 10 ratio, with no frames from the same video belonging to both the training and validation sets at the same time. Since only a single video contained a cow with a BCS of 2.00, the frames from that video were removed from the BCS analysis, resulting in 9,319 frames being used for training and validation, and 2,310 being used for testing.

The BCS classification model was constructed based on a ResNet-50 network (He et al., 2016) pretrained on the ImageNet dataset (Deng et al., 2009), and the output layer and loss function were defined following the Consistent Rank Logits (**CORAL**) framework (Cao et al., 2020) for rank-consistent ordinal regression. The target classes represented the 10 different BCS measures in quarter-point increments from 2.25 to 4.50, excluding BCS of 2.00 because only one video contained that value. The training procedure and hyperparameters were the same as used in the image quality classification model, with an additional image augmentation step performed to randomly jitter brightness, contrast, and saturation by 40% each. The performance of the trained model was assessed by evaluating the accuracy for predicting the exact BCS quarter-point values, as well as the accuracies considering error tolerances of 0.25, 0.50, 0.75, and 1.0.

Sensor and Management Data

Cows that were not present in the genotypic database or that did not have all three videos recorded before calving during individual weighing were excluded from further subclinical ketosis prediction analysis. The remaining 89 cows had the following information retrieved from the management software at the farm: parity, days in milk of the previous lactation, previous dry period length, number of past ketosis events, and weekly BCS in the last three weeks before calving. Electronic roughage intake control bins (Hokofarm Group; Marknesse, the Netherlands)

measured the weight and duration of all meals from these animals between 21 days before the expected calving date and the actual calving date. The daily averages of dry matter intake (**DMI**), feeding time, and average meal duration were calculated for both 7 days and 2 days prior to the calving date for each cow. Additional behavioral data were collected via SMARTBOW (Zoetis; Kalamazoo, MI) ear tags fitted to each cow, including lying time, rumination time, and time spent inactive and highly active, and daily averages were computed for the last 7 days before the calving date. This resulted in 20 management software and sensor variables per cow, as illustrated in Table 5.1.

Subclinical Ketosis Prediction

Models for predicting postpartum subclinical ketosis were trained and tested using prepartum data from 89 cows, including genotypic, imaging, management software, and sensor data. Those multiple data modalities were merged for analysis using different data fusion techniques, including cooperative learning, which were separately evaluated. Genotypic information was represented by 128 UMAP features, body shape information was represented by features extracted from depth frames using the BCS classification network, cow history information was collected from the management software at the farm, and behavior information was represented by descriptive statistics calculated from data collected from different sensors installed at the farm.

From each video collected weekly from 21 to 7 days before the expected calving date, 10 random frames were extracted and processed, and 2,048 features were extracted from the second-to-last layer of the BCS classifier for each frame. Because BCS is a partially subjective measurement, extracting features from the trained classifier can be a more objective and detailed way of quantifying the body shape of dairy cows. Extracting features from the 10 random frames

of each video resulted in 30 sets of 2,048 features per cow, which were then organized into 10 sets of 6,144 features by concatenating the features from frames of the three consecutive prepartum weeks. Therefore, each cow contained 128 genotypic features, 20 management software and sensor features, and 10 variations of 6,144 image features, resulting in 10 data points per cow and 6,292 features per data point. Having multiple data points per cow allowed multiple sets of frames to be used for training and testing the SCK prediction models without resorting to averaging the features extracted from different images of the same cow video and potentially losing information. Table 5.1 includes descriptions of all the features representing each cow for SCK prediction.

The target variable of the subclinical ketosis predictors was the highest BHB value measured for each cow from the blood samples collected from 1 to 14 days after calving. Four data fusion techniques were evaluated: early fusion, which consists of simply concatenating all features from different modalities before training the model; simple late fusion, which consists of separately training one model for each modality and averaging the individual predictions to achieve a final prediction; ordinary least squares (**OLS**) late fusion, which consists of separately training one model for each modality and then training an OLS regressor using the individual predictions on the training set to achieve a final predictor; and cooperative learning, which introduces an agreement penalty to encourage predictions from different modalities to agree, and chooses the degree of agreement in an adaptive manner through cross-validation (Ding et al., 2022). An overview of the data fusion techniques explored in this study is illustrated in Figure 5.1. Additionally, we assessed how removing dry matter intake measurements from the analysis impacted the results, as dry matter intake can be a difficult variable to measure in commercial settings without intake control bins. Finally, we evaluated performing principal component

analysis (**PCA**) on the genotype and image modalities before training the models to match the same number of components as management software and sensor features for each modality.

For each fusion technique, regressors with the least absolute shrinkage and selection operator (**LASSO**) penalty were trained using 70 cows, and hyperparameter tuning was performed using grid search and 5-fold cross-validation within those cows. After the optimal values were found for the LASSO regularization and the cooperative learning agreement constants, the models were retrained using such values and the full training set containing 70 cows. The models were tested on the remaining 19 cows, which were part of the predetermined testing cows used in previous analyses for genotypic and imaging feature extraction and BCS classification. After training the models, the target values and predictions were converted to binary values for SCK classification evaluation using a BHB threshold of 1.0 mmol/L for considering SCK events. This resulted in 30 out of 70 and 7 out of 19 cows having SCK in the training and testing sets, respectively. The performance metrics used to compare the different data fusion techniques were the mean absolute error (**MAE**) of BHB prediction, and the accuracy, precision, recall, specificity, and $F_1$ score of SCK classification, which is the harmonic mean of the precision and recall.

It is important to highlight that a single, consistent testing set of cows was chosen to assess the performance of the SCK prediction model and all image processing models except for animal identification. This exception was because the animal identification model utilized a closed-set supervised learning approach, which required all animals to be present in both the training and testing sets. The decision to use a single set of testing cows, rather than generating multiple random testing sets, was driven by the need to evaluate the overall performance of the cloud computing pipeline, as only a single trained model could be deployed for each data processing module. This approach aimed to mimic the scenario of introducing a new set of cows into the herd for BCS

evaluation and early detection of SCK, triggering the entire data processing pipeline from genotypic, imaging, and behavioral feature extraction to the final SCK prediction. In this scenario, only the animal identification model would need retraining and all other modules could be reused for evaluating the newly introduced cows.

Cloud Computing Pipeline

All steps of the genotypic, imaging, and sensor data processing and feature extraction, as well as animal identification, BCS, and subclinical ketosis predictive models, were deployed to a modular cloud computing pipeline hosted in Microsoft Azure and based on Representational State Transfer (**REST**) Application Programming Interfaces (**API**). Each API function was implemented as a serverless Azure Function triggered under different conditions, such as a Hypertext Transfer Protocol (**HTTP**) call or when data became available in certain Azure Blob Storage containers, designed to store large amounts of unstructured data such as documents and media files. Using this modular approach allows for the re-use of certain core services, such as body segmentation and image quality assessment, and facilitates the implementation and deployment of new functionalities into the cloud platform, all seamlessly to the farm operations as no updates are required in the on-premises farm computer infrastructure.

For genotypic data, two functions were implemented as Microsoft Azure Functions: *ProcessGenotypeRef* and *ExtractFeaturesGenotype*. The *ProcessGenotypeRef* function is triggered by an HTTP GET request with no parameters and reads all genotype files stored in an Azure Blob Storage container (*genfilesref*) following the format made available by CDCB. These files contain the 78,964 SNP values of the animals meant to be used as a reference for further SNP quality control and UMAP training – 144 animals for the current case study. The function stores in a Structured Query Language (**SQL**) table (*SNPs*), for each SNP, its predominant value and

whether it passes the quality control criteria. After replacing missing values with their corresponding SNP predominant value, the function then automatically trains a UMAP model and saves its parameters to another Azure Blob Storage container for future use (*models*). The *ExtractFeaturesGenotype* function is then triggered every time a new genotype file is available at an Azure Blob Storage container (*genfiles*). That container is supposed to receive files from new animals that are not part of the reference dataset and from which it is desired to extract genotype features and perform further phenotype predictions. The function removes invalid SNPs, replaces missing values with their corresponding predominant SNP values, and performs inference using the previously trained UMAP model, resulting in 128 features per cow. These features are stored in an SQL table (*FeaturesGenotype*) for further analysis.

For processing imaging data, seven functions were implemented as Microsoft Azure Functions: *DetectAnimal*, *ClassifyGoodBad*, *IdentifyAnimal*, *PredictBCS*, *ExtractFeaturesImage*, *CheckImageAvailability*, and *ProcessImage*. The *DetectAnimal* function is triggered by an HTTP POST request, wherein the parameter is an 8-bit depth image processed as previously described, and it returns the corresponding mask predicted by the cow body segmentation model. The *ClassifyGoodBad* function is triggered by an HTTP POST request that contains a segmented, rotated, and cropped depth image as parameter and returns whether that image is *good* or *bad* using the image quality classification model. The *IdentifyAnimal* function is triggered by an HTTP POST request that contains a segmented, rotated, and cropped infrared image as parameter and returns the cow tag identification number predicted by the cow identification model. The *PredictBCS* function is triggered by an HTTP POST request containing a segmented, rotated, and cropped depth image as parameter and returns the BCS predicted by the BCS classification model. The *ExtractFeaturesImage* function is triggered by an HTTP POST request containing a segmented,

rotated, and cropped depth image as parameter and returns the 2,048 features extracted from the second-to-last layer of the BCS classification model.

The *CheckImageAvailability* and *ProcessImage* functions work as orchestrators that call the other image processing functions when new images become available in the corresponding Azure Blob Storage container (*images*). The *CheckImageAvailability* function is triggered every 10 minutes and reads all blobs in an Azure Blob Storage container (*images*) containing depth and infrared images to be processed. The images should be named *<camera_id>_<timestamp>_<suffix_and_extension>*, with the *camera_id* being a text identifier for the camera that captured that image, the *timestamp* being the time when the image was captured in the format *yyyymmddHHMMss*, and the *suffix_and_extension* being equal to *_d.tif* for depth images, and *_i.png* for infrared images. The depth and infrared images corresponding to the same snapshot should have the same *camera_id* and *timestamp*. The function then stores in an SQL table (*ImagesAvailable*) whether the depth and infrared images are available in the container for each *camera_id* and *timestamp* combination. If both images are available and they had not been processed before, the function calls the *ProcessImage* function for that combination of *camera_id* and *timestamp*. This ensures that the pipeline implemented in *ProcessImage* is only activated when both depth and infrared images are available in the Azure Storage Blob container (*images*) for the corresponding snapshot. The *ProcessImage* function is triggered by an HTTP POST request containing as parameters the *camera_id* and *timestamp* of a snapshot to process. The function loads both the depth and infrared images corresponding to the *camera_id* and *timestamp* combination from the Azure Blob Storage container (*images*) and performs the following procedures: (1) it processes the depth image as previously described and calls *DetectAnimal*, which returns a predicted segmentation mask of the cow contained in that image; (2) it segments, rotates, and crops

the depth image using the predicted segmentation mask and calls *ClassifyGoodBad*, which returns whether that image is *good* or *bad*; (3) if the image is classified as *bad*, it ends the pipeline for that snapshot, otherwise it calls the *PredictBCS* and *ExtractFeaturesImage* functions using the segmented, rotated, and cropped depth image, and calls the *IdentifyAnimal* function using the segmented, rotated, and cropped infrared image; (4) it stores in an SQL table (*IdentificationAndBCS*) the predicted cow identification number and BCS for that snapshot; and (5) it stores in an SQL table (*FeaturesImage*) the 2,048 features returned by the *ExtractFeaturesImage* function call for that snapshot.

The *ExtractFeaturesSensor* Azure Function was implemented for processing management software and sensor data. This function is triggered every time a new file is available at an Azure Blob Storage container (*sensorfiles*). The files sent to this container should follow the comma-separated values (**CSV**) format and contain either cow history data, electronic roughage intake control bin data, or behavior data. The history data files contain, for each cow, the seven values retrieved from the farm management software described in Table 5.1. The electronic roughage intake control bin data files contain the duration in minutes, dry matter intake in kilograms, the date, and the corresponding cow identification number for all meals computed by the intake control system. The behavior data files contain, for each cow and date, the time in minutes spent lying, ruminating, inactive, and highly active. The files should be named <*prefix*>_<*file_id*>.*csv*, where *prefix* is equal to *hist* for history data files, *intake* for intake control bin data files, and *activity* for activity data files, and *file_id* is a numerical unique identifier for that file. The function reads the contents of the file and calculates the corresponding values for the wearable sensors and management software features shown in Table 5.1. It then stores the calculated values for each cow in an SQL table (*FeaturesSensor*) for further analysis.

Finally, the *PredictKetosis* Azure Function was implemented for performing subclinical ketosis prediction for a given cow using the features previously extracted and stored in SQL tables (*FeaturesGenotype*, *FeaturesImage*, and *FeaturesSensor*). It is triggered by an HTTP GET request containing as a parameter the cow identification number of the cow for which it is desired to perform a subclinical ketosis prediction. The function first reads a CSV file named *calving_dates*.csv from an Azure Blob Storage container (*supplemental*) containing the calving dates of each cow. It then finds all image keys (combination of *camera_id* and *timestamp*) for which the cow of interest was predicted via the *IdentificationAndBCS* SQL table. It groups those image keys based on the number of weeks before the calving date of the corresponding cow and builds up to 10 different sets of three image keys corresponding to three, two, and one week before the calving date. The function then retrieves the image features from the *FeaturesImage* SQL table and builds up to 10 sets of 6,144 features using the corresponding sets of image keys. For the genotype features, the function simply reads the 128 features for that cow from the *FeaturesGenotype* SQL table, and for the management software and sensor features, the function reads the *FeaturesSensor* SQL table. The function then loads the best SCK prediction model from an Azure Blob Storage container (*models*) and performs inference on up to 10 variations of 6,292 features for that cow. The median BHB value predicted for those up to 10 data points is then calculated, and the function returns *1* if that value is above the 1.0 mmol/L threshold or *0* otherwise, representing whether the system predicts that the evaluated cow has a high risk of developing subclinical ketosis postpartum. Using the median of the BHB predictions causes the function to output the most common binary-converted SCK prediction for each cow. If there are no genotype or sensor features available for that cow, or if there are no image features available for three, two,

or one week before the calving date of that cow, the function returns an error stating that it was not possible to perform a subclinical ketosis prediction because features were missing for that cow.

## RESULTS AND DISCUSSION

The objectives of this study were to propose a cloud computing-based framework for phenotypic and genotypic data processing and integration, and to evaluate different data fusion strategies for the early detection of postpartum SCK in dairy cows using wearable sensors, imaging systems, and genotypic data. Since it was not part of the objectives of this study to compare different machine learning methods or perform comprehensive hyperparameter searches for each image processing model, the performance metrics for each deep neural network are only reported once per testing set.

### Image Processing Models

Deep neural networks were trained and evaluated for four image processing tasks: cow body segmentation, image quality classification, animal identification, and BCS classification.

The cow body segmentation model achieved Dice similarity coefficients of 0.944 and 0.990 on the *test_seg_lane* and *test_seg_scale* testing sets, respectively. This indicates that the model was very effective at segmenting cow bodies in both images automatically taken at milking parlor exit lanes containing lactating cows, and especially in images manually taken at the scale containing prepartum cows, which are the ones used for the early detection of SCK.

The image quality classifier achieved accuracy, precision, recall, and specificity of 92.9%, 86.6%, 99.7%, and 87.3% on the *test_quality_lane* testing set, and 99.1%, 98.5%, 99.8%, and 98.5% on the *test_quality_scale* testing set. This shows that the model could classify as *good* 99.7% and 99.8% of the *good* lane and scale test images, missing very few *good* images overall,

and it could correctly discard from the pipeline 87.3% and 98.5% of the *bad* lane and scale test images, being especially good at detecting *bad* images taken on the scale.

Of the 20,170 images collected at the milking parlor exit lanes for this study, 12,368 were considered *bad* and 7,802 were considered *good*. This means that only under 40% of the images collected contained a single cow without any occlusion on its body (classified as *good*) and would consequently provide reliable results in further analyses. If the other 60% of the images (classified as *bad*) were not filtered by the image quality classification model, they would potentially generate incorrect results in downstream tasks, as parts of the cow body might be occluded, or multiple cows were captured in the same image. Processing those poor-quality images not only leads to unreliable results in the system but also impacts the storage and computing resource requirements of the cloud computing platform. In other words, an effective image quality classification model prevents unreliable results from being generated from poor-quality images and saves storage and computing resources in the cloud by preventing bad images from being unnecessarily processed. This image quality assessment step is crucial for automated image processing pipelines in scenarios where it is not possible to control the animal posture, lighting conditions, object occlusion, and other conditions that might impact the image quality and, consequently, the final predictions resulting from the system.

The animal identification model achieved an accuracy of 93.2% on the testing set containing images of the last four videos taken from each cow. This shows that this model trained using six distinct short videos of each cow can identify which of the 90 cows is present in new images with good accuracy, comparable to other methods reported in the literature with a similar number of cows (Zhao et al., 2019; Xiao et al., 2022; Ferreira et al., 2023). Animal identification is an essential component of any automated individual phenotyping system, and its performance

is critical for the whole pipeline to generate relevant results. Misidentified animals could lead to incorrect phenotype predictions; thus, it is imperative that the animal identification model performs with good accuracy. It is worth noting that six of those 90 cows were completely black, making them harder to differentiate, and the model achieved an accuracy of 95.1% on the other 84 cows. For cows that have very similar coat color patterns, including those belonging to breeds that do not present as many color pattern variations as Holstein, it is also possible to perform individual identification using depth images taken from a top-down view of their back (Ferreira et al., 2022), differentiating individuals based exclusively on their body shape. Additionally, open-set animal identification techniques have been proposed to identify new individuals as they are introduced to the herd without the need to retrain the deep neural networks from scratch (Andrew et al., 2021; Wang et al., 2024). The modularized nature of the proposed image processing pipeline allows for such methods to replace or complement the currently implemented animal identification module without affecting the rest of the pipeline.

The BCS classification model performed, on the testing set, 35.0%, 81.1%, 96.2%, 99.1%, and 99.7% of the predictions within 0, 0.25, 0.50, 0.75, and 1.00 points of the considered true values, which were the closest value in quarter-point increments to the average BCS assessed by the three evaluators. It achieved an MAE of 0.222 on the testing set, also considering quarter-point increments. This model achieved accuracies comparable to those previously reported in the literature, especially considering the approaches that also have a high level of automation (Qiao et al., 2021). As reported in previous works, because BCS is partially a subjective evaluation, it is difficult to achieve very high accuracies when considering exact matches between the model output and the human observation. This is why it is important to evaluate how the model performs when tolerating minor deviations between 0.25 and 1.0. As shown in Figure 5.7, the model tends

to predict more mild values for the BCS extremes, but it can seemingly differentiate between thin and fat cows, and it does not make major mistakes. This indicates that the model might have learned how to estimate a rough body shape for the cows through BCS, which is essential for its effective use as a feature extractor. Although the quality of BCS prediction might not have a direct correlation with the quality of the model as a feature extractor, the fact that it does not make major mistakes is a good indicator that it might be able to extract relevant features related to cow body shape from the depth images.

Subclinical Ketosis Prediction

For evaluating subclinical ketosis prediction, we compared five different models: *Early*, *LateSimple*, *LateOLS*, and *Coop*, corresponding to the four data fusion techniques previously described and illustrated in Figure 5.1; and *Desc_sensor*, which corresponds to a LASSO regressor using only the 20 predictive variables that originated from management software and wearable sensor data, described in Table 5.1. For each model, we evaluated reducing the dimensionality of the genotype and image modalities to 20 components each using PCA, and explored the impact of removing dry matter intake features as predictive variables, as that can be a challenging piece of information to acquire in large-scale commercial farms. We evaluated the BHB regression performance using the MAE of the predictions, and we analyzed the subclinical ketosis binary classification performance by converting the BHB regression predictions to binary values using the 1.0 mmol/L threshold and assessing classification accuracy, precision, recall, specificity, and $F_1$ score. The best BHB regression performance (MAE = 0.242) was achieved when performing early fusion (*Early*) or cooperative learning (*Coop*) and not including dry matter intake data as predictive variables. The best SCK binary classification performance ($F_1$ score = 0.750) was achieved when performing late fusion coupled with OLS regression (*LateOLS*) or when simply

using only the management software and wearable sensor variables as predictors. Performing PCA did not significantly impact the overall performance of the models.

The subclinical ketosis models performed considerably better when including dry matter intake measurements as predictors (when considering binary classification), and performing PCA before training the models did not significantly impact their predictive performance, as shown in Figures 5.8 and 5.9. Cooperative learning and early fusion resulted in identical models for all analyses, due to the optimal agreement penalty term found for cooperative learning being equal to zero in all cases. As previously described (Ding et al., 2022), cooperative learning can be especially powerful when the different data modalities are correlated and all modalities contain signal in relation to the target variable. Conversely, when modalities are uncorrelated and one of them contains more signal than the others, cooperative learning tends to be equivalent to early fusion. This seems to be the case in this study, as genotypic, imaging, management software, and wearable sensor data are seemingly not highly correlated, and cow history and behavioral data appear to have higher signal in relation to BHB than the other modalities.

In most cases, the early fusion and cooperative learning techniques achieved the lowest BHB MAE values (0.242, 0.258, 0.251, and 0.256 following the order of the bar plots in Figure 5.8), while OLS late fusion or using just the management software and sensor data achieved the highest $F_1$ scores (0.625, 0.625, 0.750, and 0.750 following the order of the bar plots in Figure 5.9). While cooperative learning and early fusion output BHB predictions that are, on average, closer to the target values (reflected by the lower MAE), they make more mistakes than OLS late fusion for classifying cows that will have low or high BHB values after calving and would therefore be exposed to a lower or higher risk of developing SCK postpartum (reflected by the lower classification performance).

All OLS late fusion models had the coefficients related to the image and genotype separate models equal to zero, meaning that the cow history and behavioral (sensors) data were better predictors when used alone than when including the image and genotype features. If those coefficients were not equal to zero, the separate image and genotype predictors would hinder the final late fusion model, as reflected also by the fact that the simple late fusion models, which output the average of the separate models' predictions, performed poorly in all analyses. This indicates, again, that the genotype and image features did not contain a strong signal in relation to the postpartum BHB measurements. This is possibly a consequence of the relatively low number of animals used to train the genotype and image feature extractors, as those modalities have very high dimensionality and consequently usually require large amounts of varied data to extract meaningful information from. In addition, due to the relatively small number of genotyped animals, we have utilized only the simplest method for missing genotype imputation, and more sophisticated methods should be evaluated in future studies, such as those taking into consideration linkage disequilibrium and pedigree information, for example (Marchini and Howie, 2010). Thus, the fact that the features extracted from those modalities were not very good predictors for postpartum BHB in this study does not necessarily mean that they are not related. Studies including a larger number of animals are essential for further exploring this phenomenon, as previous research shows that imaging and genomics data can be good contributors for disease detection in humans (Bodalal et al., 2019). Moreover, the ketosis prediction models used in this study are linear (LASSO), which might not be able to capture the non-linear relationships between some of the features and the target variable. Future work could explore the use of non-linear machine learning models such as Random Forest and Artificial Neural Networks in combination with cooperative learning and other data fusion techniques for subclinical ketosis prediction.

Although including genotype and image features through cooperative learning resulted in lower regression error (MAE) in all but one case (performing PCA and not including dry matter intake, where *Desc_sensor* performed better), the best SCK classification performance was achieved by using just the cow history and behavioral features (*LateOLS* and *Desc_sensor*). However, when analyzing the recall of the trained models, using cooperative learning always resulted in better or at least the same performance as the models relying on just cow history and behavioral features (0.771 versus 0.714 without PCA and without including dry matter intake; and 0.857 for all models when including dry matter intake), which highlights the potential benefits of including other data modalities such as images and genomics. This result is notable because, when performing early detection of SCK, a high recall is very important for preventing sick cows from going undetected by the model, as false negatives, in this case, are usually more costly than false positives. In other words, ignoring a cow that will eventually become sick and potentially lead to larger losses is generally more costly than treating a cow unnecessarily trying to prevent SCK (Cainzos et al., 2022). Previous research (McArt et al., 2015; Steeneveld et al., 2020) estimates that a single case of subclinical ketosis can cost, on average, between $171 and $289 to the dairy farmer, reaching up to $1,365 in some extreme cases of clinical ketosis. Moreover, treating cows with propylene glycol can have great economic benefits (up to $1,166 per 100 fresh cows), especially if the only ones being treated are those that tested positive for hyperketonemia from 3 to 9 days in milk (McArt et al., 2014). Being able to detect in advance the cows that have a greater risk of developing SCK facilitates the adoption of more focused and cost-effective treatment strategies. With this early detection being performed in a fully automated and non-intrusive manner via the proposed cloud computing framework, dairy farmers can not only obtain significant

economic benefits but also improve animal health and welfare by implementing preventive actions against hyperketonemia in dairy cows.

Dairy cows are complex organisms, and the prevalence of subclinical ketosis is affected by many different factors, some of which were not considered in this study, such as management and nutritional practices and diet composition (Duffield, 2000). With the modular approach implemented in the proposed cloud computing framework, other sources of information can be included into the data processing and SCK prediction pipeline, potentially enhancing its predictive performance. In addition, other predictive models can be implemented by reutilizing parts of the existing pipeline, as it can already extract information from genotypic, behavioral, and body shape imaging data, which are associated with many other phenotypes that are useful for dairy farming.

Another benefit of implementing this modular framework in a cloud computing platform is that each part of the pipeline can be altered and optimized with the development of new algorithms and methodologies. In this case study, for example, not only the model that predicts BCS can be improved and subsequently updated within the pipeline, but also each feature extraction function can be altered to include other feature extraction methods such as autoencoders, foundation models, and other self-supervised techniques. Additionally, in a scenario where not all data modalities might be available simultaneously, having separate models trained using a single modality or combinations of modalities allows for more flexibility in using the modalities as they become available. For instance, an initial prediction could be performed using only the available genotypic data for a new animal in the herd, and as images are collected and data from wearable sensors become available, a more robust prediction would be performed using the best data fusion technique for those modalities.

With the rapid advancement of artificial intelligence algorithms, adopting a modular approach is key for future-proofing the proposed system and facilitating its use and improvement. Implementing a framework based on reusable modules facilitates the development of new tools and functionalities within the cloud computing ecosystem, which contributes to both the advancement of scientific research and the emergence of new PLF solutions that support management decisions in livestock farming and, ultimately, animal health and welfare.

## CONCLUSION

The results reported in this study show that, when dry matter intake is available, OLS late fusion is the best model for classification. However, in the absence of dry matter intake measurements, cooperative learning, despite yielding a lower $F_1$ score due to increased false positives, exhibits lower MAE and, more importantly, higher recall compared to OLS late fusion, and thus is the optimal model in that case. A higher recall means that fewer high-risk cows go undetected and thus untreated, enabling the adoption of more focused and cost-effective treatment strategies for hyperketonemia in dairy farms. Implementing the proposed automated system for the early detection of subclinical ketosis in dairy cows could not only drastically reduce the negative economic effects of peripartum hyperketonemia, but also improve animal health and welfare. Furthermore, the modularized and multimodal nature of the proposed framework facilitates the enhancement of current feature extractors and phenotype predictors, as well as the development of new predictive models and functionalities into the cloud computing system, allowing for an ecosystem of PLF solutions for improving management decisions in dairy farms.

## ACKNOWLEDGMENTS

The authors would like to thank the USDA National Institute of Food and Agriculture (Washington, DC; grant 2023-68014-39821/accession no. 1030367) for the financial support.

## REFERENCES

Allaoui, M., M.L. Kherfi, and A. Cheriet. 2020. Considerably Improving Clustering Algorithms Using UMAP Dimensionality Reduction Technique: A Comparative Study. Pages 317–325 in Image and Signal Processing. Springer International Publishing, Cham.

Alonso, R.S., I. Sittón-Candanedo, Ó. García, J. Prieto, and S. Rodríguez-González. 2020. An intelligent Edge-IoT platform for monitoring livestock and crops in a dairy farming scenario. Ad Hoc Networks 98:102047. doi:https://doi.org/10.1016/j.adhoc.2019.102047.

Andrew, W., J. Gao, S. Mullan, N. Campbell, A.W. Dowsey, and T. Burghardt. 2021. Visual identification of individual Holstein-Friesian cattle via deep metric learning. Comput Electron Agric 185:106133. doi:https://doi.org/10.1016/j.compag.2021.106133.

Atrey, P.K., M.A. Hossain, A. El Saddik, and M.S. Kankanhalli. 2010. Multimodal fusion for multimedia analysis: a survey. Multimed Syst 16:345–379. doi:10.1007/s00530-010-0182-0.

Baltrušaitis, T., C. Ahuja, and L.-P. Morency. 2019. Multimodal Machine Learning: A Survey and Taxonomy. IEEE Trans Pattern Anal Mach Intell 41:423–443. doi:10.1109/TPAMI.2018.2798607.

Becht, E., L. McInnes, J. Healy, C.-A. Dutertre, I.W.H. Kwok, L.G. Ng, F. Ginhoux, and E.W. Newell. 2019. Dimensionality reduction for visualizing single-cell data using UMAP. Nat Biotechnol 37:38–44. doi:10.1038/nbt.4314.

Berckmans, D. 2017. General introduction to precision livestock farming. Animal Frontiers 7:6–11. doi:10.2527/af.2017.0102.

Bodalal, Z., S. Trebeschi, T.D.L. Nguyen-Kim, W. Schats, and R. Beets-Tan. 2019. Radiogenomics: bridging imaging and genomics. Abdominal Radiology 44:1960–1984. doi:10.1007/s00261-019-02028-w.

Brito, L.F., H.R. Oliveira, B.R. McConn, A.P. Schinckel, A. Arrazola, J.N. Marchant-Forde, and J.S. Johnson. 2020. Large-Scale Phenotyping of Livestock Welfare in Commercial Production Systems: A New Frontier in Animal Breeding. Front Genet 11.

Cainzos, J.M., C. Andreu-Vazquez, M. Guadagnini, A. Rijpert-Duvivier, and T. Duffield. 2022. A systematic review of the cost of ketosis in dairy cattle. J Dairy Sci 105:6175–6195. doi:https://doi.org/10.3168/jds.2021-21539.

Cao, W., V. Mirjalili, and S. Raschka. 2020. Rank consistent ordinal regression for neural networks with application to age estimation. Pattern Recognit Lett 140:325–331. doi:https://doi.org/10.1016/j.patrec.2020.11.008.

Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. Pages 248–255 in 2009 IEEE Conference on Computer Vision and Pattern Recognition.

Ding, D.Y., S. Li, B. Narasimhan, and R. Tibshirani. 2022. Cooperative learning for multiview analysis. Proceedings of the National Academy of Sciences 119:e2202113119. doi:10.1073/pnas.2202113119.

Duffield, T. 2000. Subclinical Ketosis in Lactating Dairy Cattle. Veterinary Clinics of North America: Food Animal Practice 16:231–253. doi:https://doi.org/10.1016/S0749-0720(15)30103-1.

ElKarami, B., A. Alkhateeb, H. Qattous, L. Alshomali, and B. Shahrrava. 2022. Multi-omics Data Integration Model Based on UMAP Embedding and Convolutional Neural Network. Cancer Inform 21:11769351221124204. doi:10.1177/11769351221124205.

Fernandes, A.F.A., J.R.R. Dórea, and G.J. de M. Rosa. 2020. Image Analysis and Computer Vision Applications in Animal Sciences: An Overview. Front Vet Sci 7.

Ferreira, R.E.P., T. Bresolin, G.J.M. Rosa, and J.R.R. Dórea. 2022. Using dorsal surface for individual identification of dairy calves through 3D deep learning algorithms. Comput Electron Agric 201:107272. doi:https://doi.org/10.1016/j.compag.2022.107272.

Ferreira, R.E.P., Y.J. Lee, and J.R.R. Dórea. 2023. Using pseudo-labeling to improve performance of deep neural networks for animal identification. Sci Rep 13:13875. doi:10.1038/s41598-023-40977-x.

He, K., X. Zhang, S. Ren, and J. Sun. 2016. Deep residual learning for image recognition. Pages 770–778 in Proceedings of the IEEE conference on computer vision and pattern recognition.

Hinton, G., N. Srivastava, and K. Swersky. 2012. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. Cited on 14:2.

Holdorf, H.T., S.J. Kendall, K.E. Ruh, M.J. Caputo, G.J. Combs, S.J. Henisz, W.E. Brown, T. Bresolin, R.E.P. Ferreira, J.R.R. Dorea, and H.M. White. 2023. Increasing the prepartum dose of rumen-protected choline: Effects on milk production and metabolism in high-producing Holstein dairy cows. J Dairy Sci 106:5988–6004. doi:https://doi.org/10.3168/jds.2022-22905.

Keselman, L., J.I. Woodfill, A. Grunnet-Jepsen, and A. Bhowmik. 2017. Intel(R) RealSense(TM) Stereoscopic Depth Cameras. Pages 1267–1276 in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).

Kingma, D.P., and J. Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Koltes, J.E., J.B. Cole, R. Clemmens, R.N. Dilger, L.M. Kramer, J.K. Lunney, M.E. McCue, S.D. McKay, R.G. Mateescu, B.M. Murdoch, R. Reuter, C.E. Rexroad, G.J.M. Rosa, N.V.L. Serão, S.N. White, M.J. Woodward-Greene, M. Worku, H. Zhang, and J.M. Reecy. 2019. A Vision for Development and Utilization of High-Throughput Phenotyping and Big Data Analytics in Livestock. Front Genet 10.

Marchini, J., and B. Howie. 2010. Genotype imputation for genome-wide association studies. Nat Rev Genet 11:499–511. doi:10.1038/nrg2796.

McArt, J.A.A., D. V Nydam, G.R. Oetzel, and C.L. Guard. 2014. An economic analysis of hyperketonemia testing and propylene glycol treatment strategies in early lactation dairy cattle. Prev Vet Med 117:170–179. doi:https://doi.org/10.1016/j.prevetmed.2014.06.017.

McArt, J.A.A., D. V Nydam, and M.W. Overton. 2015. Hyperketonemia in early lactation dairy cattle: A deterministic estimate of component and total cost per case. J Dairy Sci 98:2043–2054. doi:https://doi.org/10.3168/jds.2014-8740.

McInnes, L., J. Healy, and J. Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426.

Neethirajan, S., and B. Kemp. 2021. Digital Livestock Farming. Sens Biosensing Res 32:100408. doi:https://doi.org/10.1016/j.sbsr.2021.100408.

Qiao, Y., H. Kong, C. Clark, S. Lomax, D. Su, S. Eiffert, and S. Sukkarieh. 2021. Intelligent perception for cattle monitoring: A review for cattle identification, body condition score evaluation, and weight estimation. Comput Electron Agric 185:106143. doi:https://doi.org/10.1016/j.compag.2021.106143.

Ronneberger, O., P. Fischer, and T. Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. Pages 234–241 in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Springer International Publishing, Cham.

Schokker, D., M. Poppe, J. ten Napel, I.N. Athanasiadis, C. Kamphuis, and R.F. Veerkamp. 2022. Rapid turnover of sensor data to genetic evaluation for dairy cows in the cloud. J Dairy Sci 105:9792–9798. doi:https://doi.org/10.3168/jds.2022-22113.

Silva, F.F., G. Morota, and G.J. de M. Rosa. 2021. Editorial: High-Throughput Phenotyping in the Genomic Improvement of Livestock. Front Genet 12.

Steeneveld, W., P. Amuta, F.J.S. van Soest, R. Jorritsma, and H. Hogeveen. 2020. Estimating the combined costs of clinical and subclinical ketosis in dairy cows. PLoS One 15:e0230448-.

Wang, R., R. Gao, Q. Li, C. Zhao, L. Ru, L. Ding, L. Yu, and W. Ma. 2024. An ultra-lightweight method for individual identification of cow-back pattern images in an open image set. Expert Syst Appl 249:123529. doi:https://doi.org/10.1016/j.eswa.2024.123529.

Wildman, E.E., G.M. Jones, P.E. Wagner, R.L. Boman, H.F. Troutt, and T.N. Lesch. 1982. A Dairy Cow Body Condition Scoring System and Its Relationship to Selected Production Characteristics. J Dairy Sci 65:495–501. doi:https://doi.org/10.3168/jds.S0022-0302(82)82223-6.

Xiao, J., G. Liu, K. Wang, and Y. Si. 2022. Cow identification in free-stall barns based on an improved Mask R-CNN and an SVM. Comput Electron Agric 194:106738. doi:https://doi.org/10.1016/j.compag.2022.106738.

Zhao, K., X. Jin, J. Ji, J. Wang, H. Ma, and X. Zhu. 2019. Individual identification of Holstein dairy cows based on detecting and matching feature points in body images. Biosyst Eng 181:128–139. doi:https://doi.org/10.1016/j.biosystemseng.2019.03.004.

Zou, K.H., S.K. Warfield, A. Bharatha, C.M.C. Tempany, M.R. Kaus, S.J. Haker, W.M. Wells, F.A. Jolesz, and R. Kikinis. 2004. Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports. Acad Radiol 11:178–189. doi:https://doi.org/10.1016/S1076-6332(03)00671-8.

# TABLES AND FIGURES

**Table 5.1.** Description of the features extracted from genotypic data, wearable and imaging sensor data, and management software information from each individual cow for the early detection of postpartum subclinical ketosis. DRTC stands for days respective to calving, UMAP stands for Uniform Manifold Approximation and Projection, and BCS stands for body condition score.

| Modality | Features | DRTC | Variations per cow | Description |
|---|---|---|---|---|
| Genotype | 128 UMAP features | | 1 | 128 dimensions resulting from UMAP model trained using binary SNPs values after quality control. |
| Image | 2,048 neural network features | -21 | 10 | 10 sets of 2,048 features extracted from the second-to-last layer of the BCS classification model using 10 depth frames collected at 21 days before calving. |
| | 2,048 neural network features | -14 | 10 | 10 sets of 2,048 features extracted from the second-to-last layer of the BCS classification model using 10 depth frames collected at 14 days before calving. |
| | 2,048 neural network features | -7 | 10 | 10 sets of 2,048 features extracted from the second-to-last layer of the BCS classification model using 10 depth frames collected at 7 days before calving. |
| Management software | 4 parity dummy variables | | 1 | 4 one-hot encoded dummy variables representing cow parity. Second lactation cows were encoded as 0000, third lactation as 0001, and so on. |
| | Previous days in milk | | 1 | Number of days in milk in the previous lactation. |
| | Previous days dry | | 1 | Number of days dry between previous and current lactations. |
| | Ketosis events | | 1 | Total number of ketosis events in previous lactations. |
| | BCS -21 | -21 | 1 | BCS assessed 21 days before expected calving date. |
| | BCS -14 | -14 | 1 | BCS assessed 14 days before expected calving date. |
| | BCS -7 | -7 | 1 | BCS assessed 7 days before expected calving date. |
| Wearable sensors | Intake -7 | -7 to -1 | 1 | Average daily dry matter intake during the last 7 days prior to calving. |
| | Intake -2 | -2 to -1 | 1 | Average daily dry matter intake during the last 2 days prior to calving. |
| | Feeding time -7 | -7 to -1 | 1 | Average daily time spent feeding during the last 7 days prior to calving. |
| | Feeding time -2 | -2 to -1 | 1 | Average daily time spent feeding during the last 2 days prior to calving. |
| | Meal duration -7 | -7 to -1 | 1 | Average meal duration during the last 7 days prior to calving. |
| | Meal duration -2 | -2 to -1 | 1 | Average meal duration during the last 2 days prior to calving. |
| | Lying time -7 | -7 to -1 | 1 | Average daily time spent lying during the last 7 days prior to calving. |
| | Rumination time -7 | -7 to -1 | 1 | Average daily time spent ruminating during the last 7 days prior to calving. |
| | Inactive -7 | -7 to -1 | 1 | Average daily time spent inactive during the last 7 days prior to calving. |
| | Highly active -7 | -7 to -1 | 1 | Average daily time spent highly active during the last 7 days prior to calving. |
| | Highly active -7 | -7 to -1 | 1 | Average daily time spent highly active during the last 7 days prior to calving. |

**Table 5.2.** Description of all Azure Functions implemented in the proposed cloud computing framework.

| Modality | Function | Trigger | Description |
|---|---|---|---|
| Genotype | *ProcessGenotypeRef* | HTTP GET | Reads reference genotype files, performs quality control, and trains UMAP model. |
| | *ExtractFeaturesGenotype* | File available in Blob Storage container | Reads new genotype file available, performs quality control based on reference values, extracts features using trained UMAP model, and saves them to an SQL table. |
| Image | *DetectAnimal* | HTTP POST | Returns the cow segmentation mask predicted from a processed depth image received in the request body. |
| | *ClassifyGoodBad* | HTTP POST | Returns whether the segmented depth image received in the request body is *good* or *bad.* |
| | *IdentifyAnimal* | HTTP POST | Returns the cow identification number predicted from the segmented infrared image received in the request body. |
| | *PredictBCS* | HTTP POST | Returns the BCS predicted from the segmented depth image received in the request body. |
| | *ExtractFeaturesImage* | HTTP POST | Returns the 2,048 features extracted from the segmented depth image received in the request body. |
| | *CheckImageAvailability* | Every 10 minutes | Reads all files available in a Blob Storage container to be processed. If both infrared and depth images are available for a certain camera ID and timestamp, calls *ProcessImage.* |
| | *ProcessImage* | HTTP POST | Reads the infrared and depth images available for the received camera ID and timestamp and runs the pipeline, calling each of the other corresponding functions. Saves cow identification number, BCS classification, and 2,048 image features to SQL tables. |
| Management software and wearable sensors | *ExtractFeaturesSensor* | File available in Blob Storage container | Reads management software or wearable sensor files and saves calculated features to SQL table. |
| - | *PredictKetosis* | HTTP GET | Returns SCK prediction for the received cow identification number. Returns descriptive error if missing modality. |

**Table 5.3.** Performance of each image processing model on the independent testing sets.

| Model | Metric | Performance at scale | Performance at lane |
|---|---|---|---|
| Cow body segmentation | Dice similarity coefficient | 0.990 | 0.944 |
| Image quality classifier | Accuracy | 99.1% | 92.9% |
| Animal identifier | Accuracy | 93.2% | - |
| BCS classifier | Accuracy @ 0, 0.25, 0.50 tolerance | 35.0%, 81.1%, 96.2% | - |

**Table 5.4.** Performance of the different evaluated models without performing PCA. Including genotype and image features through cooperative learning or early fusion resulted in lower regression error (MAE), but the highest $F_1$ score was achieved by using just the cow history and behavioral features (LateOLS and Desc_sensor). Using cooperative learning or early fusion always resulted in better or at least the same recall as the models relying on just cow history and behavioral features, meaning that they predict fewer false negatives, which are generally more costly in the context of SCK detection.

| Include DMI? | Models | MAE | Accuracy | $F_1$ score | Precision | Recall | Specificity |
|---|---|---|---|---|---|---|---|
| **Yes** | *Coop* and *Early* | 0.251 | 71.6% | 0.690 | 0.577 | 0.857 | 0.633 |
| | *LateOLS* | 0.340 | 78.9% | 0.750 | 0.667 | 0.857 | 0.750 |
| | *Desc_sensor* | 0.277 | 78.9% | 0.750 | 0.667 | 0.857 | 0.750 |
| **No** | *Coop* and *Early* | 0.242 | 60.5% | 0.590 | 0.478 | 0.771 | 0.508 |
| | *LateOLS* | 0.326 | 68.4% | 0.625 | 0.556 | 0.714 | 0.667 |
| | *Desc_sensor* | 0.255 | 63.2% | 0.588 | 0.500 | 0.714 | 0.583 |

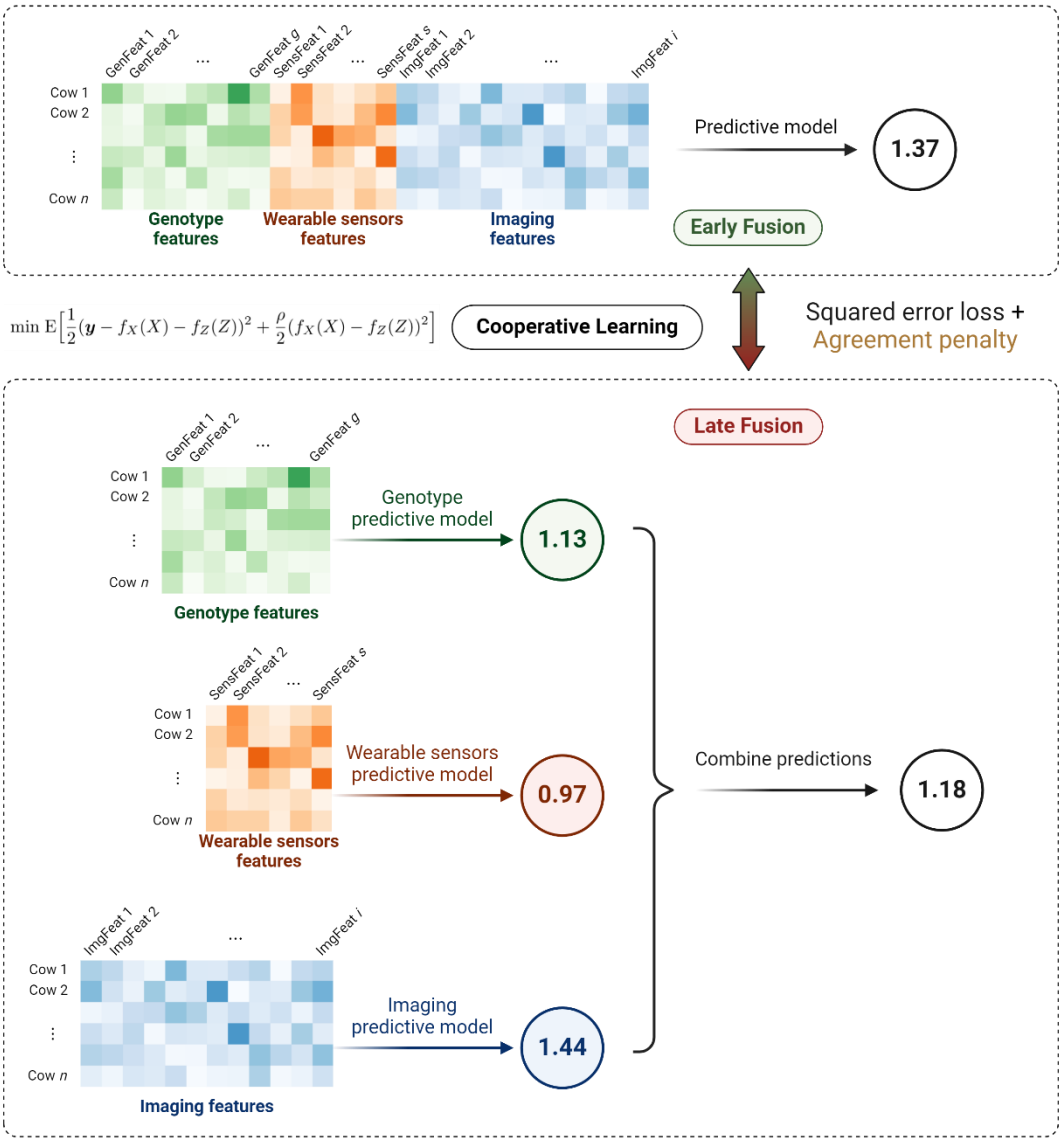



**Figure 5.1.** Overview of the multimodal data fusion techniques explored in this study. Adapted from (Ding et al., 2022). In early fusion, features from all modalities are combined prior to training a predictive model. In late fusion, each modality has its own separate predictive model, and the separate predictions are combined into a final prediction. Cooperative learning is a hybrid of the two approaches, introducing an agreement penalty that encourages predictions from different modalities to agree, resulting in a spectrum of potential solutions ranging from early to late fusion methods. The level of agreement is chosen in a data-adaptive manner through cross-validation to minimize validation error. In this study, the three different modalities explored were genotypic data, images, and data collected from wearable sensors and farm management software. The target variable of the predictive models was the concentration of plasma beta-hydroxybutyrate in mmol/L, which indicates subclinical ketosis.

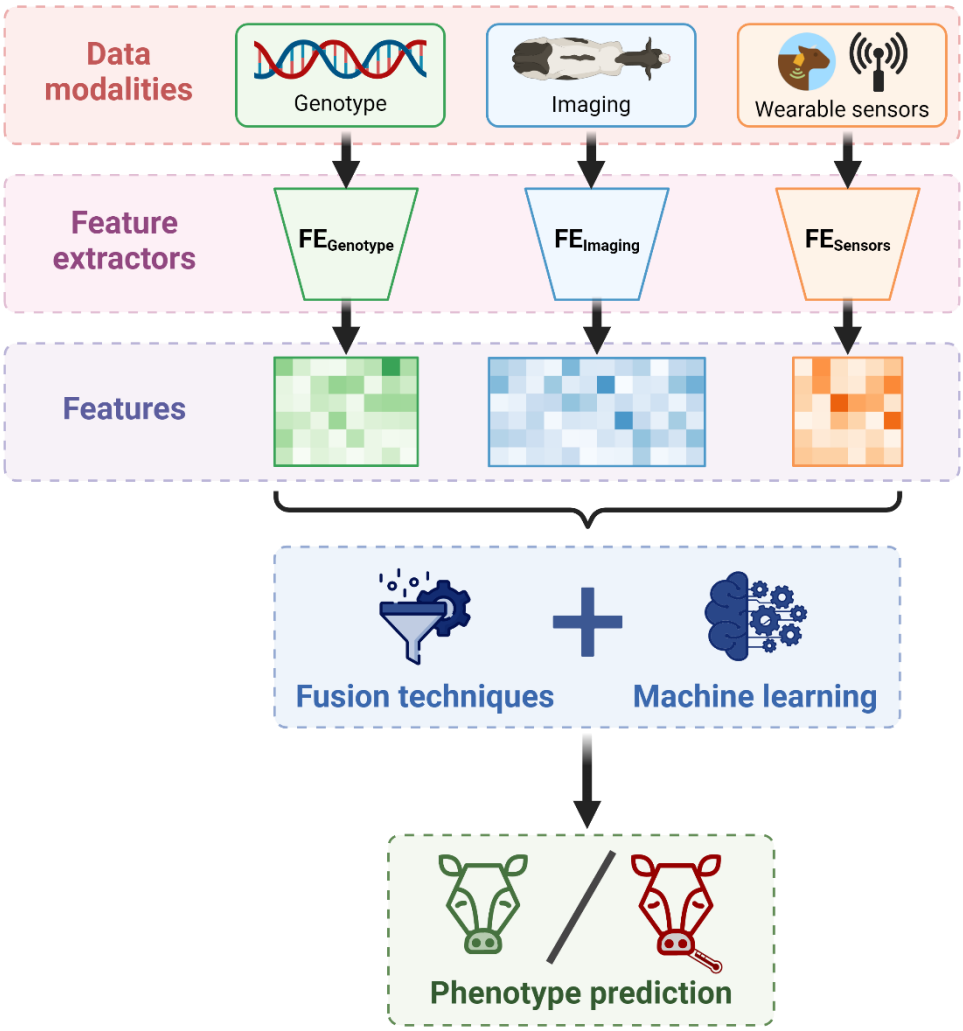



**Figure 5.2.** Overview of the subclinical ketosis prediction pipeline implemented in this study. Features are extracted from each data modality (genotype, imaging, and wearable sensors) and data fusion techniques are applied to the extracted features for phenotype prediction using machine learning algorithms. In this study, the target phenotype is the early detection of postpartum subclinical ketosis through plasma BHB concentration prediction.

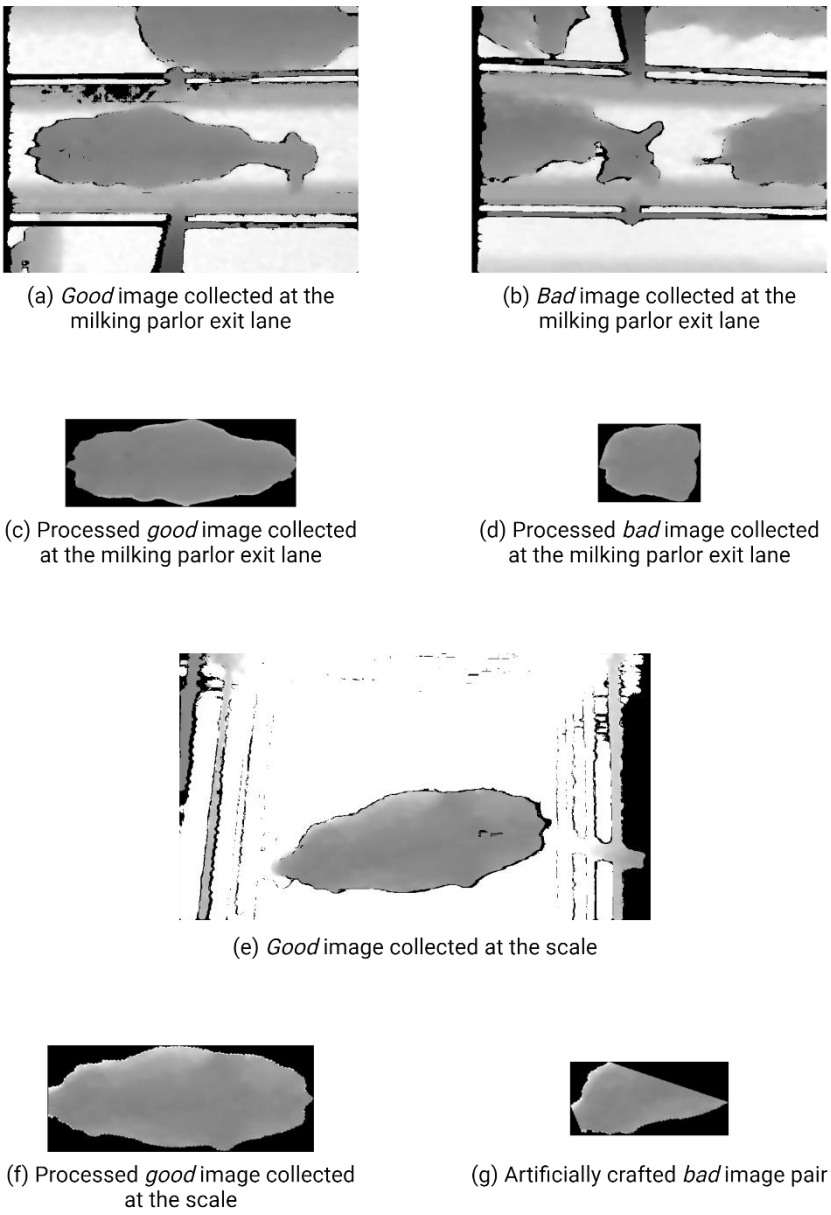



**Figure 5.3.** Examples of good and bad images collected automatically at the milking parlor exit lane (a, b, c, and d) and manually at the scale (e, f, and g). Since images at the scale were collected manually, there were no bad examples, so those were artificially crafted to simulate situations where the cow body would be partially occluded (g). Images a, b, and e illustrate the collected depth images before any processing except for pixel normalization for better visualization. Images c, d, f, and g illustrate the preprocessed depth images that were used to train, validate, and test the image quality classification model.

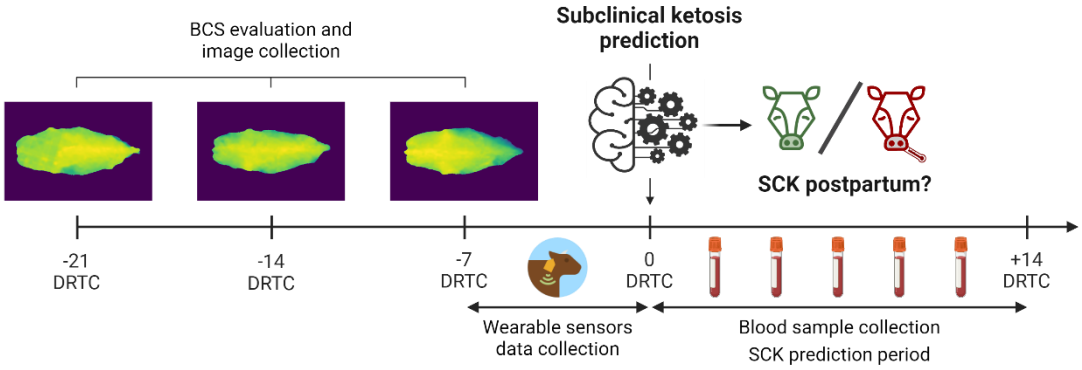



**Figure 5.4.** Data collection timeline. BCS evaluation and depth image collection were performed weekly during the three last weeks before the calving date, data from wearable sensors were collected during the last week before the calving date, and blood samples were collected during the two weeks following the calving date. Machine learning models were trained using genotypic data and prepartum depth images, wearable sensor data, and information extracted from the farm management software, to predict cases of subclinical ketosis postpartum. Since only prepartum data were used to train the models, the predictions can be performed at the calving date, enabling early detection of subclinical ketosis.

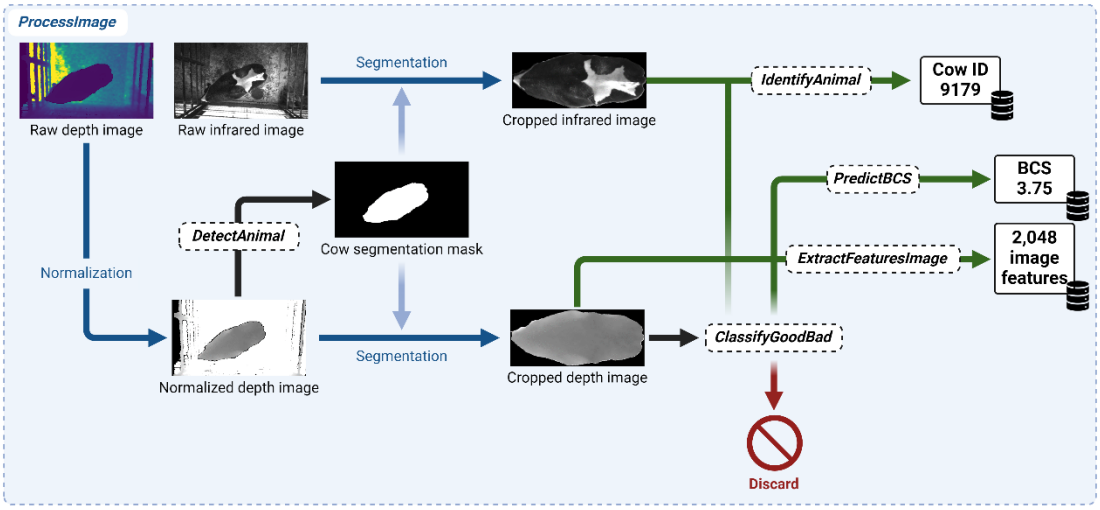



**Figure 5.5.** Image processing and feature extraction procedures implemented in the cloud computing pipeline. Dashed boxes with font in bold and italic represent Azure functions and solid boxes with font in bold represent the values returned by those functions. Arrows and text in blue represent actions performed by the *ProcessImage* orchestrator function. As both depth and infrared images are available, *ProcessImage* is called. *ProcessImage* normalizes the depth image and passes it as input to the cow body segmentation model via the *DetectAnimal* function. The predicted mask is then applied to both the depth and infrared images, and the segmented images are rotated and cropped around the cow. The cropped depth image is passed to the *ClassifyGoodBad* function and, if the image is predicted as good, the rest of the pipeline is executed, represented by green arrows. The cow identification number is predicted via the *IdentifyAnimal* function using the cropped infrared image, the BCS is predicted via the *PredictBCS* function, and the image features are extracted from the BCS classification model via the *ExtractFeaturesImage* function; they are then stored in an SQL database.

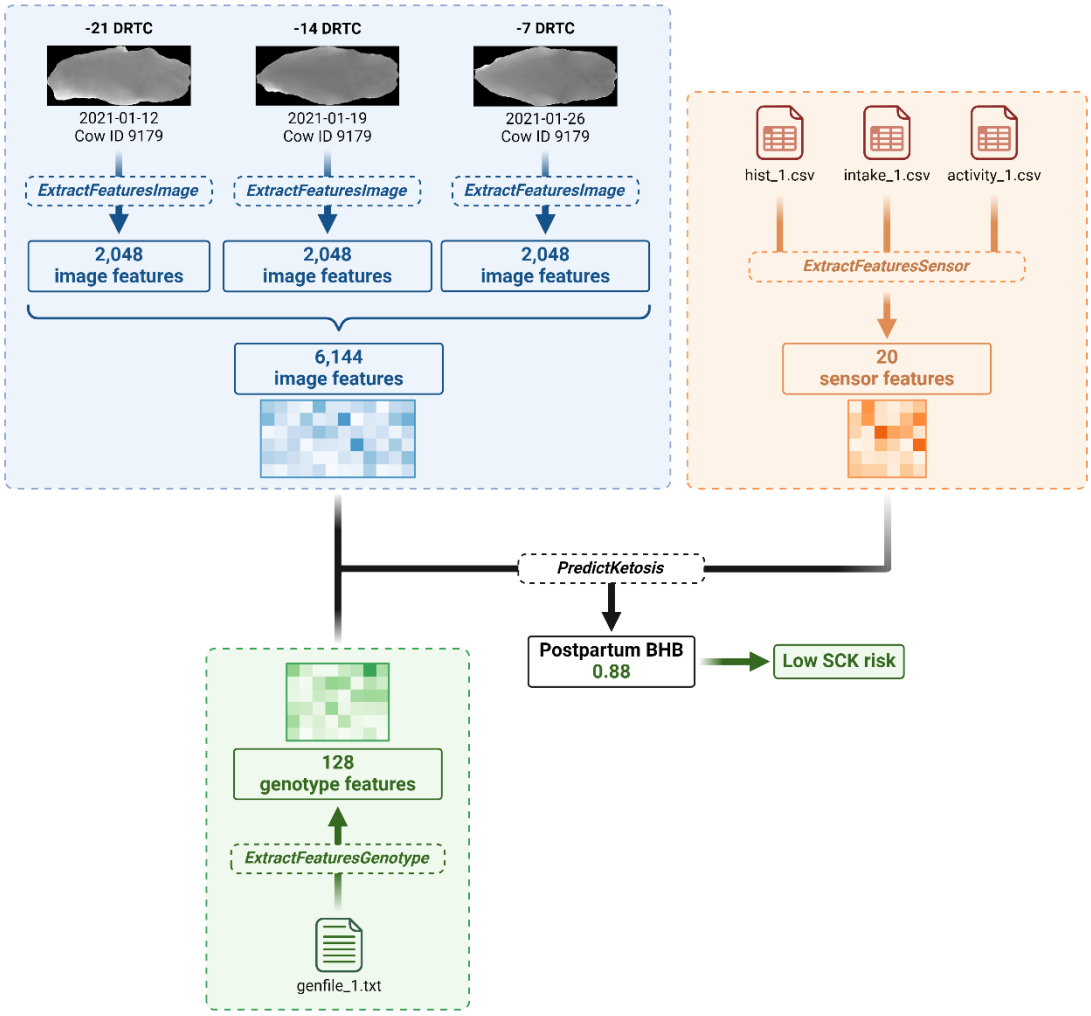



**Figure 5.6.** Feature extraction and subclinical ketosis prediction implemented in the cloud computing pipeline. Dotted boxes with font in bold and italic represent Azure functions and solid boxes with font in bold represent the values returned by those functions. Features from images collected in different weeks during the three weeks prior to the calving date are extracted and concatenated, resulting in 6,144 image features. Management software and sensor features are extracted from their corresponding CSV files, and genotype features are extracted from the genotype data files using a trained UMAP model. The 6,292 features are passed to a postpartum BHB predictor and the predicted BHB value is used to assess the risk of that cow developing subclinical ketosis postpartum by using a threshold of 1.0 mmol/L.

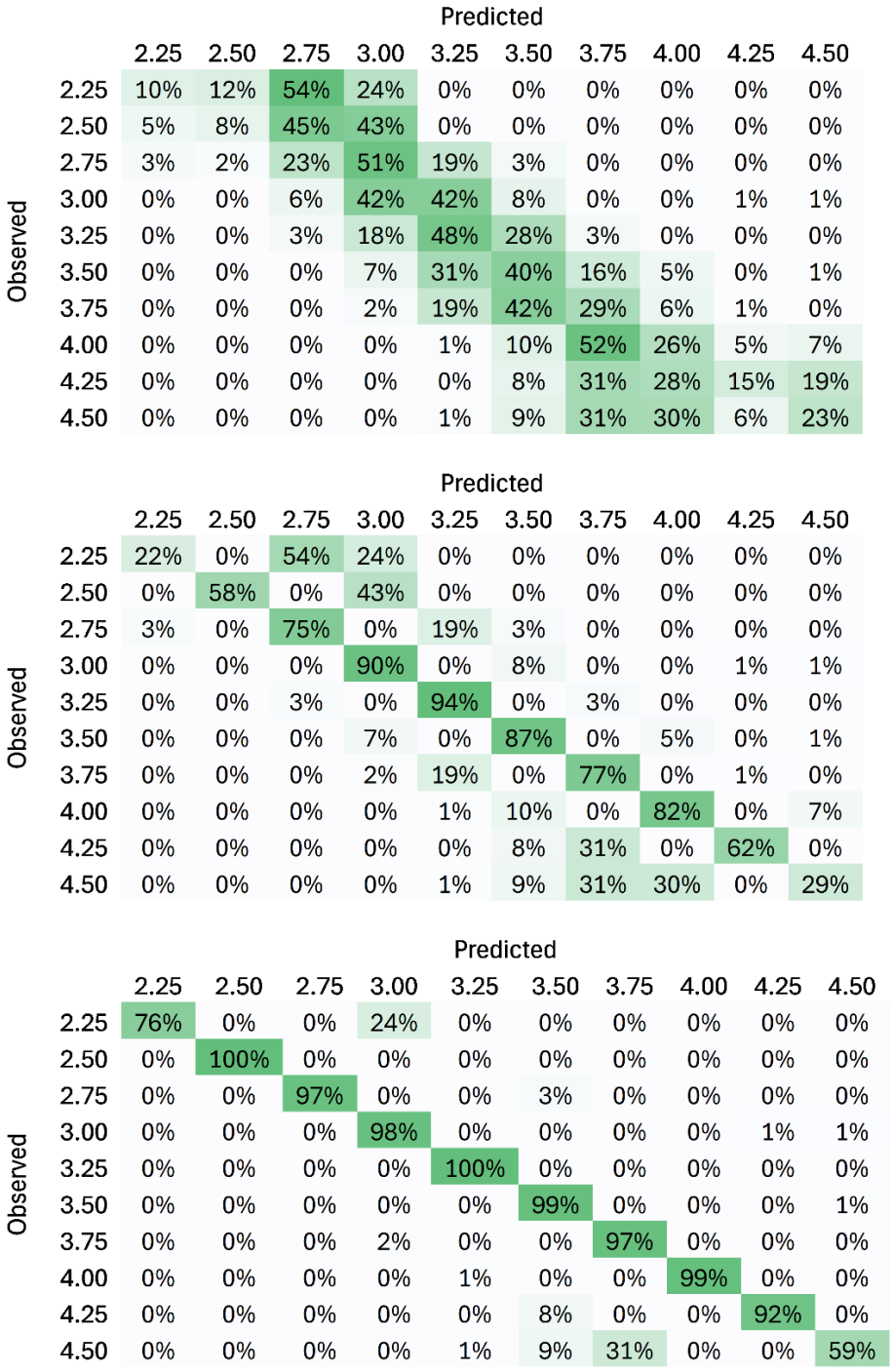



**Figure 5.7.** BCS classifier confusion matrices as percentages of the number of observed images in each class. The first matrix is the original confusion matrix without any adjustment for error tolerance (overall accuracy 35.0%), while the second and third matrices are adjusted to correct predictions with up to 0.25 and 0.50 errors, respectively, achieving accuracies of 81.1% and 96.2%. The model follows a trend of predicting mild values for the BCS extremes, but it can still seemingly differentiate between thin and fat cows, and it does not make major mistakes.

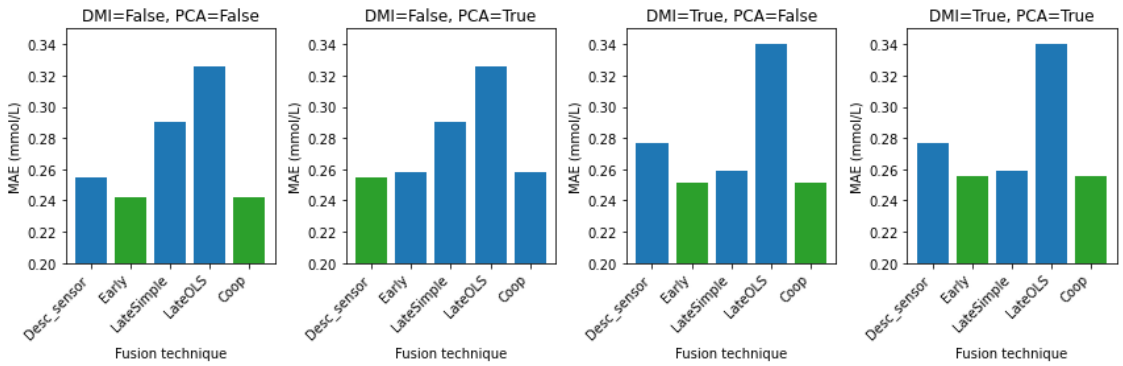



**Figure 5.8.** Testing set BHB MAE for each fusion technique and a model trained exclusively on cow history and behavioral data from sensors. The different graphs evaluate how including dry matter intake measurements and performing PCA before model training impacted on the results. Early fusion and cooperative learning achieved the best results in most cases when evaluating the BHB regression error.

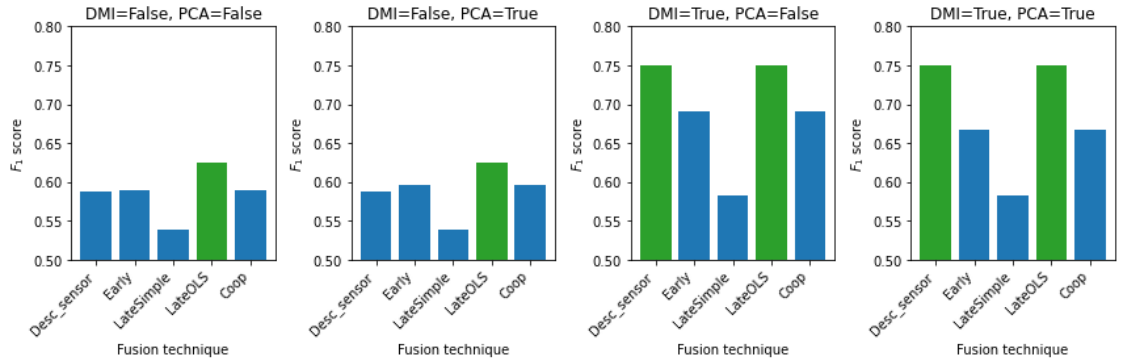



**Figure 5.9.** Testing set $F_1$ score for each fusion technique and a model trained exclusively on cow history and behavioral data from sensors. The different graphs evaluate how including dry matter intake measurements and performing PCA before model training impacted on the results. OLS late fusion and the separate cow history and behavioral model achieved the best results when evaluating the $F_1$ score classification metric.

# CHAPTER SIX: GENERAL CONCLUSIONS AND FUTURE DIRECTIONS

In livestock farming, computer vision systems (**CVS**) have the potential for predicting phenotypes in a non-intrusive way and on a large scale. The hardware used for collecting images and performing high-throughput phenotyping can be leveraged for individual animal identification using computer vision algorithms. Several methods have been proposed for identifying dairy cows based on unique biometrics such as muzzle patterns, iris, facial features, and coat color patterns. The most promising methods for commercial implementation are those that recognize coat color patterns, as they only require top-down camera views from a far enough distance, which can be easily achieved by installing cameras in convenient locations at the farm that are far from animal reach. However, such methods can only be applied to animal breeds that exhibit uniquely identifiable color patterns, such as Holstein cows. Only a few studies have explored the potential of identifying animals using 3-dimensional (**3D**) representations, which could be applicable to any breed. Nevertheless, to the best of our knowledge, the study presented in Chapter 2 was the first to validate an animal identification CVS as the animals experience changes in their body shape due to growth. The results of this study suggest that the methods evaluated are able to learn unique biometrics from the 3D surface of the calves that remain recognizable even as their body size changes due to growth. However, this experiment was done using only five animals and larger datasets should be collected and evaluated in future studies.

The deep neural networks explored in previous animal identification studies typically require large amounts of annotated data to generalize well across different environments, which can be costly and time-consuming to obtain. In Chapter 3, a semi-supervised learning technique called pseudo-labeling was explored for improving the performance of deep neural networks for animal identification, while requiring less annotated data. The method evaluated in this study

complements current animal identification research by seamlessly integrating with existing models, requiring no retraining or modifications to the existing architecture. Future research in CVS for animal identification should focus on the open-set problem, as commercial farms have dynamic herds with new animals being constantly added. Additionally, applying semi-supervised learning techniques to these open-set models could enhance their performance and reduce data annotation requirements.

In Chapters 4 and 5, computer vision and machine learning techniques were explored for the early detection of subclinical ketosis (**SCK**) in dairy cows by integrating data from different sources. In Chapter 4, methods for extracting features from high-dimensional unstructured data such as images and text were evaluated within the machine learning pipeline for SCK prediction. This study represented a first attempt at extracting body shape information from depth images for performing early detection of health issues in dairy cows. Moreover, this study was also pioneering in extracting information from text data using modern NLP techniques and integrating it into dairy cow phenotype predictive models. The superior performance achieved by integrating image and text features into the SCK prediction highlights the potential of leveraging unstructured data, that would otherwise be difficult to analyze, for early disease detection and phenotype prediction. The feature extraction, data processing, and machine learning pipelines proposed in this study can be applied to other phenotypes for exploration in future studies, further enhancing the decision-making process in dairy farms to reduce costs and improve productivity and animal health.

Inspired by the promising results achieved through the integration of different types of data into a machine learning pipeline for SCK prediction, the study presented in Chapter 5 proposed a cloud-computing framework to automate and facilitate access to feature extraction and phenotype prediction algorithms. This framework was applied to SCK prediction using genotype, imaging,

behavior, and cow historical data, and multiple data fusion techniques were evaluated for integrating these different data modalities. The proposed framework was designed to be easily expanded by implementing an approach based on modules that perform different data processing steps independently. This allows the implemented modules to be re-used for the prediction of other phenotypes that are related to genetics, body shape, and behavior. Furthermore, this approach facilitates the improvement of currently implemented modules and the development of new modules that process data in different ways. In other words, the cloud-computing framework proposed in Chapter 5 has the potential to contribute to advancements in scientific research related to the development of precision livestock farming (**PLF**) tools aimed at enhancing the productivity and efficiency of livestock farms via improved data-drive decision-making processes.