ESTRUS DETECTION IN COWS WITH DEEP LEARNING TECHNIQUES

A Thesis Submitted to
the Graduate School of Engineering and
Sciences of İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in Computer Engineering

by
İbrahim ARIKAN

MAY 2024
İZMİR

We approve the thesis of **İbrahim ARIKAN**

Examining Committee Members:

Prof. Dr. Tolga AYAV
Department of Computer Engineering, Izmir Institute of Technology

Asst. Prof. Fatih SOYGAZİ
Department of Computer Engineering, University of Aydın Adnan Menderes

Assoc. Prof. Dr. Ahmet Çağdaş SEÇKİN
Department of Computer Engineering, University of Aydın Adnan Menderes

Prof. Dr. Tuğkan TUĞLULAR
Department of Computer Engineering, Izmir Institute of Technology

Asst. Prof. Emrah İNAN
Department of Computer Engineering, Izmir Institute of Technology

27 May 2024

Prof. Dr. Tolga AYAV
Supervisor, Department of Computer Engineering,
İzmir Institute of Technology

Asst. Prof. Fatih SOYGAZİ
Co-supervisor, Department of Computer Engineering,
University of Aydın Adnan Menderses

Prof. Dr. Cüneyt Fehmi BAZLAMAÇCI
Head of Computer Engineering Department

Prof. Dr. Mehtap EANES
Dean of the Graduate School of Engineering and Sciences

ACKNOWLEDGEMENTS

First of all, I would like to thank my mother, my father and my older sister, who were with me from the day I was born to my primary school years, then throughout my secondary school, high school and university life, for adding to my life and supporting me unconditionally during this master's degree period. I am grateful to them for always being with me throughout this process.

I would like to thank the esteemed my supervisor Prof. Dr. Tolga AYAV, whom I met at the beginning of my master's studies and who has never ceased to show interest and provide assistance ever since. Throughout our collaboration, I have continuously learned new knowledge from him.

And, I want to express my gratitude to the respected my co-supervisor Asst. Prof. Fatih SOYGAZİ, whom I met during my undergraduate education and had the opportunity to work with for many years, benefited from his experiences and encouraged me throughout the time we worked together, enabling me to take our research work to further levels.

And finally, I would like to thank the esteemed Assoc. Prof. Ahmet Çağdaş SEÇKİN, who helped me identify my shortcomings in the preparation of this study.

Prof. Dr. Tolga AYAV and Asst. Prof. Fatih SOYGAZİ, with their feedback and advice from my first research days to the results, helped me achieve better results and contributed to my development as a beginner researcher.

ABSTRACT

ESTRUS DETECTION IN COWS WITH DEEP LEARNING TECHNIQUES

Accurately predicting the estrus period is essential for enhancing the efficiency and lowering the costs of artificial insemination in livestock, a crucial sector for global food production. Precisely identifying the estrus period is critical to avoid economic losses such as decreased milk production, delayed calf births, and loss of eligibility for government subsidies.

Since the most obvious movement that needs to be detected during the fertilization period is mounting, it is important to detect this movement. Since manual detection of this movement is difficult and costly, automated methods were needed. Therefore, it is thought that deep learning-based methods can be applied to detect the mounting moment.

The proposed method detects the estrus period using deep learning and XAI (Explainable Artificial Intelligence) techniques. Deep learning-based mounting detection is performed using CNN, ResNet, VGG-19 and YOLO-v5 models. The ResNet model in this proposed study detects mounting movement with 99% accuracy.

Explainability of deep learning models describes features that aid in decision-making in detecting mounting motion. Grad-CAM and Gradient Inputs models, which are XAI techniques, are used for the black box behind the proposed models. The developed deep learning models reveal that they focus on the udder and back area of the cows during the decision-making phase. In addition, how successfully the Grad-CAM and Gradient Inputs models, which are the XAI models used for the explainability of the deep learning models trained in this study, performed the explanation process was measured by calculating the "faithfulness", "maximum sensitivity" and "complexity" metrics.

ÖZET

DERİN ÖĞRENME TEKNİKLERİ İLE İNEKLERDE KIZGINLIK TESPİTİ

Doğurganlık döneminin doğru tahmini, küresel gıda üretimi için hayati bir sektör olan hayvancılıkta suni tohumlamanın verimliliğini optimize etmek ve maliyetlerini azaltmak için kritik öneme sahiptir. Süt üretiminde azalma, buzağı doğumlarının gecikmesi ve devlet desteklerinden mahrum kalma gibi ekonomik kayıpların önlenmesi için döllenme süresinin kesin olarak belirlenmesi hayati önem taşımaktadır.

Doğurganlık periyodu boyunca tespit edilmesi gereken en belirgin hareket atlama hareketi olduğu için bu hareketin tespiti önem taşımaktadır. Bu hareketin manuel tespiti zor ve maliyetli olduğu için otomatize yöntemlere ihtiyaç duyulmuştur. Dolayısıyla atlama hareketi anının tespiti için derin öğrenme tabanlı yöntemlerin uygulanabileceği düşünülmüştür.

Önerilen yöntem, derin öğrenme ve açıklanabilir yapay zeka tekniklerini kullanarak doğurganlık dönemini tespit etmektedir. Derin öğrenme tabanlı atlama tespiti CNN, ResNet, VGG-19 ve YOLO-v5 modelleri kullanılarak gerçekleştirilmektedir. Önerilen sistemdeki ResNet modeli atlama hareketini %99 doğrulukla tespit etmektedir.

Derin öğrenme modellerinin açıklanabilirliği, atlama hareketinin tespit edilmesinde karar vermeye yardımcı olan özellikleri açıklar. Önerilen modellerin arkasında yer alan kara kutu için açıklanabilir yapay zeka tekniklerinden olan Grad-CAM ve Gradient Giriş modelleri kullanılmıştır. Geliştirilen derin öğrenme modelleri, karar verme aşamasında ineklerin meme ve sırt bölgesine odaklandıklarını ortaya koymuştur. Ayrıca, bu çalışmada eğitilen derin öğrenme modellerinin açıklanabilirliği için kullanılan Grad-CAM ve Gradient Inputs gibi XAI modellerinin, 'faithfulness', 'maximum sensitivity' ve 'complexity' metrikleri hesaplanarak ölçülen açıklama işlemini ne kadar başarılı bir şekilde gerçekleştirdiği incelenmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ANN.. ……Artificial Neural Network

CNN.. Convolutional Neural Network

GMM.. Gaussian Mixtured Model

LSTM .... Long Short-Term Memory

MHI .... Movement History Image

RoI .... Region of Interest

XAI ....Explainable Artificial Intelligence

UWB ....Ultra-wideband

VGG..Visual Geometry Group

YOLO..You Look Only Once

CHAPTER 1

INTRODUCTION

The livestock industry is rapidly evolving in search of more effective, sustainable and efficient solutions to the ever-increasing demands of modern agricultural practices (Araújo et al. 2021; Liu et al. 2021). One of the main factors driving this transformation is the broad implementation of cutting-edge technologies like artificial intelligence and deep learning. Deep learning has brought about a significant transformation in livestock farming processes thanks to its ability to recognize complex patterns on large datasets. For example, analyzing a wide range of data, from animal behavior to health status, has become more precise and predictable thanks to this technology. However, the complexity brought by these technological developments has also brought with it a difficulty of understanding. The fact that deep learning algorithms are generally complex and "black box' causes their decision mechanisms to not be fully understood. At this point, XAI (Explainable Artificial Intelligence) becomes crucial, striving to clarify the decisions made by deep learning models in a comprehensible manner. In this context in the livestock sector, XAI plays a critical role in increasing the reliability of deep learning-based systems, making decision processes transparent and providing more effective interaction with the end user. Within this evolving technological landscape, livestock businesses and experts face important questions about how to effectively integrate artificial intelligence and deep learning models, as well as how to make these systems more explainable. In this context, adopting a balanced approach, both technically and ethically, is vital to ensure that AI in the agricultural sector evolves smoothly and supports sustainability.

Accurate timing in estrus detection enhances reproductive efficiency and enables effective management of genetic resources. Thus, monitoring and managing livestock is crucial for both animal welfare and production efficiency in dairy farming. The estrus period in cows is when a mature cow is most fertile and ready to be pregnant (Roelofs et al. 2010; Remnant et al. 2018). This period is often characterized by specific movements and behaviors, as illustrated in Figure 1. Cows typically have an estrus cycle lasting 21 days, and if they do not become pregnant, they will enter another estrus period approximately 21 days later. The duration of the estrus period in cows can vary based on factors such as age, seasonal conditions, and diet. Monitoring estrus in cows is crucial to ensure that conception occurs promptly following birth.

In the classification of estrus symptoms by Reith and Hoy (Reith et al., 2018), both primary and secondary signs are detailed. The most notable primary sign is "standing to be mounted," which indicates that cows are ready to mate. However, this behavior has been observed less frequently in high milk-producing cows, and its duration may also be shorter. Secondary signs include mounting behavior, increased activity, and changes in ruminating time, agonistic interactions, and social behaviors. Mounting behavior starts and continues before the primary estrus sign. The frequency of cows mounting or attempting to mount each other during the mating period is considered a more reliable indicator for detecting estrus. These signs are crucial for accurately identifying the estrus period and determining the most suitable time for artificial insemination. Missing the estrus period can lead to economic losses, such as decreased milk production, a 21-day delay in artificial insemination, and a one-month delay in calf birth (Dallago et al., 2021; Webster et al., 2020). The latest technologies in automating estrus detection include machine learning and deep learning techniques. These techniques analyze video images to detect the estrus period based on cows' activity, behavior, and/or physiological characteristics (Fricke et al., 2014).

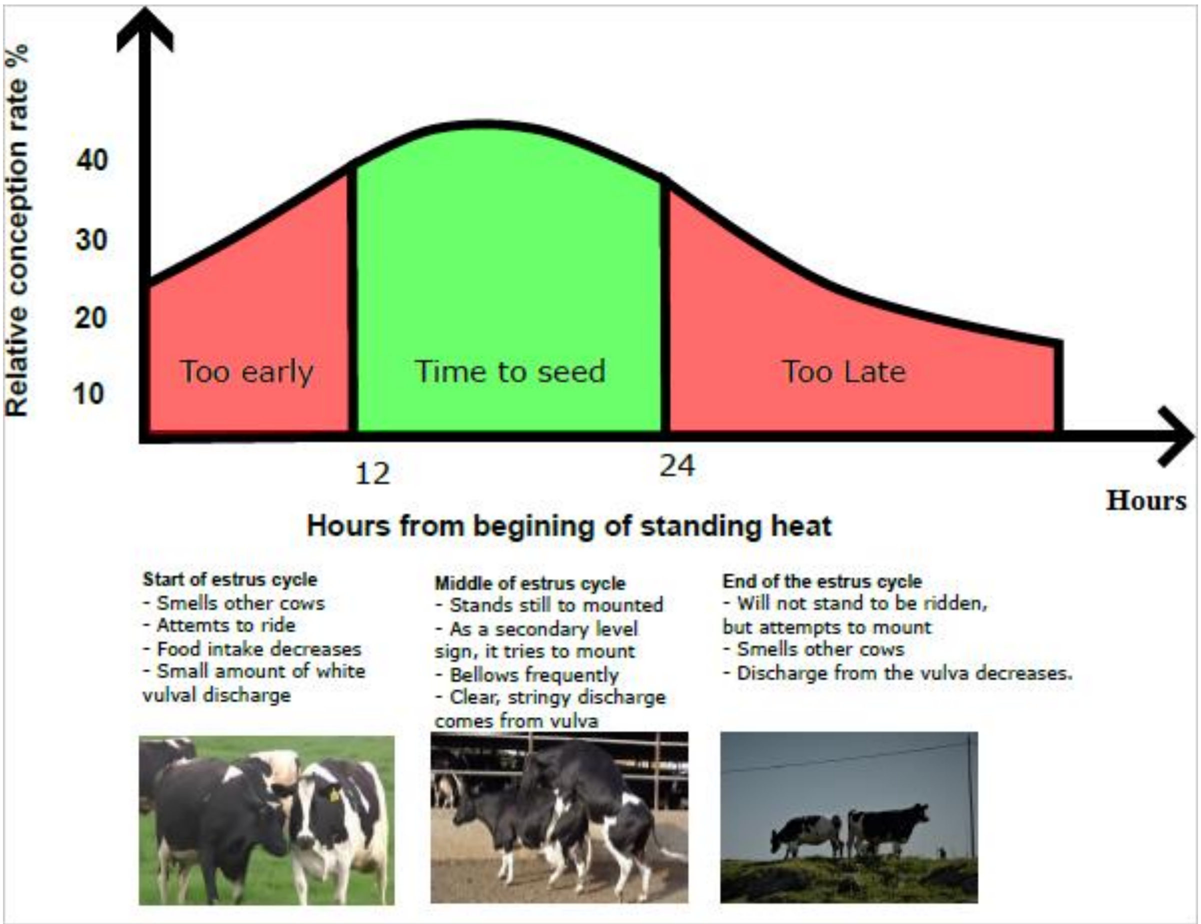

Figure 1. Estrus cycle diagram (Arıkan et al. 2023).

1.1. Related Works for Estrus Detection

When we examine the literature on the detection of estrus in cattle, Koray Yıldız's PhD thesis, the focus was on detecting estrus in cows. The aim of this study was to develop an artificial neural network (ANN) model to predict estrus in cattle, utilizing motion and environmental data (Yıldız et al. 2022). Various data sources were integrated to determine the estrus status of cows in the research, including veterinary records, climate data, cow movements, and demographic information. A wireless pedometer system was employed to track cow movements. ANN models were trained using combinations of different input types, such as motion data, climate data, and animal information, to determine the estrus status. Techniques like data normalization and validation set usage were implemented during the training process to prevent overfitting.

The optimal input variables were determined as motion data, previous period's motion data, days elapsed since the last estrus, temperature, and humidity. The ANN model created with this combination of data achieved the most successful results with a higher F-score of 0.1775 compared to other models. Additionally, the utilization of hourly data allowed for a more precise and early prediction of estrus. This study is significant in predicting the reproductive cycle of cows.

Chung et al. applied computer vision techniques using a side-view video camera to detect estrus changes in cows (Chung et al. 2015). In this way, the focus is on proposing a system to automatically detect cow mounting. Aspects of using side view cameras to address variables that challenge the accuracy and response time of typical tilted cameras. The research focused on computer vision techniques for the sudden maintenance of outages. The proposed system provides a method that can automatically detect skipping operations. They prefer to use RoI (Region of Interest), a specific region in passing scenes with the side view camera, to effectively detect mounting activities. Breaking situations can be observed at a height of 1.5 to 1.9 meters above the ground, so they placed the side view camera at a height of 1.7 meters. They introduce GMM (Gaussian Mixed Models) techniques to extract motion information accurately. This technique increases lighting performance depending on weather conditions. MHI (Movement History Image) also appears to create a movement summary for the activity. This summary table is used to accurately detect mounting. In the analysis, the direction of the vector movement is determined as a tangent graph by focusing on the change in the mounting. The beginning and ending frames must show a certain movement. This method is used to accurately detect mounting activities.

Arago et al. designed a system to identify cows' estrus by monitoring the mounting behavior exhibited during this period. Their research involved training two ANN models utilizing the TensorFlow Object Detection API, aiming to detect estrus events within a 100-meter range (Arago et al. 2020). They implemented the detection by analyzing images captured from strategically positioned cameras. The developed system achieved an accuracy of 94%.

The study conducted by S. Benaissa stands out as a significant step in the detection of calving and estrus in dairy cattle (S.Benaissa et al. 2020). The integration of accelerometers mounted on the neck and leg with ultra-wideband (UWB) indoor localization sensors allows for a more comprehensive analysis of animal behaviors and the detection of estrus periods. In this extensive research, data were gathered from 13 pregnant cows and 12 cows that had undergone successful artificial insemination. The collected data cover both pre- and post-calving estrus periods. Logistic regression based detection models created from the obtained data show that the integration of sensor systems significantly improves the performance of estrus detection. The sensitivity of 86-89% and specificity of 73-77% achieved within the specified time intervals by combining accelerometers mounted on the neck and leg highlight the effectiveness of this technology.

On the other hand, the wireless intravaginal probe introduced by Andersson presents a significant innovation aimed at automating estrus detection in cattle (Andersson et al. 2016). This probe, based on conductivity, temperature, and motion detection measurements, can accurately detect estrus symptoms. The data obtained with the probe demonstrate its compatibility with the identified estrus models.

Jun Wang emphasizes the importance of timely estrus detection in dairy farming, drawing attention to limitations in traditional methods such as background noise, data scarcity for selecting sound features, and inadequacies in voice recognition algorithms (Jun Wang et al. 2023). To address these issues, they developed a dual-channel recording device and a sound event extraction method, proving its effectiveness in noisy farm environments. Furthermore, estrus detection based on unique sound features such as consecutive vocalization count and maximum consecutive duration was optimized using statistical methods. The results of this study highlight the significant potential of sound-based technologies in improving estrus detection in dairy farming. The proposed binary LSTM discriminant strategy achieving a 100% estrus detection rate in a blind test suggests that combining sound identification with other automated detection methods could further enhance detection rates. These findings underscore the potential role of sound-based technologies and artificial intelligence algorithms in strengthening estrus detection and management in the dairy industry.

1.2. Related Works for Explainable Artifical Intelligence in Farming

The issue of explainability has not yet been addressed in artificial intelligence studies in the field of livestock farming. Alternatively, as we examine the studies on XAI in agriculture, it becomes evident that technological progress holds significant potential.

In the study by Quach et al., the tomato damage dataset was classified into categories such as Unripe, Ripe, Old, and Damaged by training deep learning models (Quach et al. 2024). The Grad-CAM XAI algorithm was employed to elucidate the results of the deep learning models. This method was used to evaluate the image recognition ability of deep learning or black box models by identifying image features in detected regions. Furthermore, to assess the integrity of the model based on Grad-CAM, the research examined all image features in the test dataset on the models, highlighting the differences in the learned features of each label. The primary objective of the research was to evaluate the reliability of the recognition model. Each trained model learned the features of the damaged area, with the DenseNet201 and NasNetMobile models effectively recognizing the background and the healthy parts of the tomato. This indicates that these models achieve a certain level of reliability for the Damage tag feature. However, for the Old tag feature, Grad-CAM revealed unreliable features, showing that the regions identified by deep learning were mixed and overlapped with the damaged area. The results from the MobileNet, ResNet50, Xception, InceptionV3, and EfficientNetV2 models failed to detect image features in the damaged fruit region. For the damage label, these models generally identified the background and edge regions of the tomato correctly, demonstrating performance similar to that for the inefficient labels. The features modeled by EfficientNetV2 and Xception were recognized accurately and were less affected by other features. Consequently, although the MobileNet model achieved the highest performance results, EfficientNetV2 and Xception were concluded to be the most reliable models.

In the study conducted by Siwar et al., a novel XAI saliency method based on specific perturbations was proposed for detecting potato diseases (Siwar et al. 2023). While previous research on explainability methods in agriculture has primarily focused on the explainability of leaf disease classification, this study introduced a perturbation-based method designed to clarify both the localization and classification aspects of potato leaf diseases. This new approach is inspired by D-RISE, the first perturbation-based method developed to explain predictions of object detections. The D-RISE method, which builds on the concept of random input masking, has limitations: random perturbations can produce crude results when creating a saliency map and require significant computational time to obtain generalized results. To overcome these challenges, the proposed method iteratively performs specific perturbations that are spatially informed by intermediate prediction results. For the training and detection of potato plant diseases, the study employed the Faster RCNN model, known for its high performance. The proposed perturbation-based method demonstrated superior performance in both metrics compared to the D-RISE method. The deletion scores obtained indicate that removing highlighted pixels based on saliency maps significantly alters the model's decisions and quickly degrades detection performance. Conversely, the insertion scores suggest that the proposed method more accurately approximates the original predictions, underscoring its effectiveness in improving the explainability and reliability of the model's results.

R. S et al. explored how plants can be afflicted by various diseases, extreme climatic conditions, and pests, which adversely impact the quality of the harvest (R. S et al. 2022). To mitigate the decline in crop yield, they aimed to surpass traditional methods by employing deep learning techniques for damage detection. They utilized Inception V3 and ResNet transfer learning models for this purpose. To understand the decision-making processes within these deep learning models, they applied Grad-CAM and LIME explainability models. The study used two primary datasets: the Plant Village dataset and the New Plant Diseases dataset. The original Plant Village dataset, used for training the ResNet model, includes 14 different crop types such as apples, potatoes, green peppers, tomatoes, blueberries, corn, cherries, grapes, peaches, soybeans, raspberries, oranges, squash, and strawberries. This dataset contains a total of 54,305 images, which represent 17 fungal diseases, 4 bacterial diseases, 2 mold diseases (oomycete), 1 disease caused by mites, and 12 healthy plant species. The ResNet model trained on this dataset achieved a 99% accuracy rate, while the Inception V3 model achieved a 95% accuracy rate. Grad-CAM and LIME explainability models were then employed to interpret the classification decisions made by the deep learning models. These explainability models helped in visualizing and understanding how the models distinguished between different types of plant diseases and healthy plants, thus providing insights into their decision-making processes.

1.3. Proposed Estrus Detection Study

The method we used to detect the estrus period in cows was determined through mounting movement with the 9-layer CNN model, VGG-19 model, YOLO-V5 model and

ResNet model. The models we trained were trained on a dataset consisting of both internet data and images we captured. These models demonstrated high accuracy rates of 98% and 99%, respectively, in detecting estrus. However, it is of great importance to understand what qualities these models base their decision-making processes on. Therefore; we tried to better understand how models make decisions by using explainability models such as Grad-Cam and Gradient Inputs for CNN, VGG-19, and ResNet models. Our explainability study was rigorously evaluated using faithfulness, complexity, and maximum sensitivity metrics. This assessment provided important information about the reliability and consistency of the statements. Of course, we can observe the high accuracy of the trained models. However, thanks to explainability analyses, we realized the necessity of working to better understand the decisions of these models and evaluate their reliability. This study was carried out using XAI methods provided by other common software for estrus detection. One of the main mathematical differences is which program focuses on determining the estrus of the developed models. In this context, we aim to ensure that estrus towards farm owners is shown in a more transparent and understandable way.

- This study aims to automate estrus detection using Convolutional Neural Network (CNN), VGG-19, YOLO-v5 and ResNet models specifically designed for estrus period detection, which is critical for cow breeding and herd management.
- Explainability models such as Grad-CAM and Gradient Inputs were used to understand the features focused on in the decision-making processes of the models. This is an important step in evaluating the success of the trained models and understanding their decisions.
- As a result of the explainability studies, the success of Grad-CAM and Gradient Inputs models was measured and compared with the "faithfulness", "maximum sensitivity" and "complexity" metrics.

- Among the deep learning models trained, the ResNet model stood out as the most successful model with a accuracy rate of 99%.
- Explainability models assist better understand the decision-making processes of trained models and reliably evaluate the success of the models.

CHAPTER 2

BACKGROUND

2.1. Deep Learning Algorithms

In this part; in our study, the history and content of studies on deep learning models used for estrus detection in cows are examined.

2.1.1. Convolutional Neural Network (CNN)

LeCun et al., in their study titled 'Gradient-Based Learning Applied to Document Recognition' published in 1998, emphasized that machine learning techniques applied to artificial neural networks play an important role in the design of pattern recognition systems (LeCun et al. 2015). This work shows that better pattern recognition systems can rely more heavily on automatic learning, reducing reliance on hand-crafted heuristics. LeCun shows that, particularly in the field of handwritten character recognition, hand-crafted feature extraction can be advantageously replaced by carefully designed learning machines that can operate directly on pixel images. In this study, while increasing the use of learning techniques, it appears that a minimum level of prior knowledge is needed for each task. In particular, specially designed artificial neural network architectures (e.g., CNN) can be used in a customized way for tasks such as handwriting recognition, specifically by containing information about the invariants of two-dimensional shapes. This study explains that there are several approaches to automatic machine learning methods, but one of the most successful is "numerical" or gradient-based learning.

For example, LeNet-5, a typical CNN used to recognize characters, takes input plane, size-normalized and centered character images. A unit in each layer receives input from a set of units in a small area in the previous layer. These local receptive areas can extract key visual features (e.g., directional edges, endpoints, corners). These features are combined in later layers and used to detect higher level features. Sharing weights is accomplished by forcing similar weight vectors on units in different locations. Each unit is connected to a set of units with local receptive fields located in different regions at the input and shares the same weight vector and the same bias. These units detect the same feature in all regions of the entrance. That is, if an input is shifted, the feature map output shifts by the same amount but remains unchanged otherwise. This property is the basis for the resilience of CNN to shifts and distortions of the input. In CNN, once a particular feature is detected, its exact location becomes less important. Only its approximate location relative to other features is important. The exact positions of these features are unnecessary to define the pattern, and it is possible that the positions will vary in different character instances. Therefore, a simple way to reduce the sensitivity of the position of salient features is to reduce the precision of encoding the positions in the feature map. This can be achieved with a subsampling layer that reduces the spatial resolution of the feature map. CNN often involve sequential convolution and subsampling operations, with layers varying in a way that continues to reduce the spatial resolution of the input and increases the richness of the representation. The combination of convolution and subsampling was inspired by Hubel and Wiesel's ideas of "simple" and "complex" cells (Hubel et al. 1962). CNN can have a structure that synthesizes its own feature extractor because all weights are learned by backpropagation.

2.1.2. Visual Geometry Group (VGG)

In 2014, Simonyan decided to examine the effect of the depth of the CNN on its accuracy in a large-scale image recognition environment (Simonyan et al. 2014). Simonyan; As a result of detailed evaluations that increased the depth of the network by using very small size (3*3) convolution filters, they observed that a 16-19 layer CNN showed improvements compared to previous studies. With the results of their evaluations, they became the most successful results in the 2014 ImageNet Challenge competition, surpassing the previously developed deep networks. As a result of their studies, they encouraged research and development by presenting the two most successful CNN models to the public as the VGG model. In this study, color images of size 224*224 were used as input data. In the architecture of the work, the image is passed through a series of convolution layers. Very small receptive fields are used using 3×3 filters. The stride of the convolution layers is fixed at 1 pixel, and spatial padding is used before each 3 × 3 convolution layer to preserve spatial resolution. Convolution layers are followed by some max-pooling layers. These layers help reduce data size. At the very end of the network are three fully connected layers. The first two layers have 4096 channels and the third layer is used for ILSVRC classification, where there are 1000 different classes. All hidden layers are equipped with a non-linear activation function called ReLU (Rectified Linear Unit). The study uses mini-batch gradient descent to optimize a multi-class logistic regression target using large-scale training images. Editing techniques such as weight reduction and dropout have also been used. In training, the initialization of weights is important, and poor initialization of weights can prevent deep networks from learning. Therefore, one starts by using a structure shallow enough that it can be randomly initialized before training deeper networks, and then some of these initial weights are used when initializing deeper structures. A trained ConvNet and an input image are used in the testing phase. It allows ConvNet to effectively classify large images while also improving computational efficiency. Although it is noted that using multiple crops may provide higher accuracy, computation time may limit this approach. Therefore, depending on the application context, the option of using dense evaluation or multiple cropping may be preferable. These methods play an important role in the development of ConvNet-based large-scale image classification systems.

2.1.3. ResNet

As neural networks become deeper, training them can become more challenging, particularly when convolution is involved. In order to address this challenge, He at al. developed a residual learning method to simplify the training of these networks (He et al. 2015). An important step they took to facilitate the optimization and training of deep neural networks was to explicitly reformulate the layers as residual functions referenced by the previous layer's inputs, rather than functions that are learned independently of prior layer inputs. Despite being eight times deeper than the VGG architecture, this study maintains lower complexity. It achieved an error rate of 3.57% on the ImageNet test set, earning it first place in the ILSVRC 2015 classification challenge. This article addresses the problem of "degradation" encountered during the training of deep neural networks. In traditional approaches, it is expected that with the stacking of several layers, the network directly learns the desired function. However, in this study, a framework that allows for the learning of a kind of "residual operation" between layers is proposed. This allows the network to focus on learning a residual operation based on the outputs of the previous layers. To express it more formally, when the desired fundamental operation is denoted as H(x), the network tries to learn a residual operation expressed as;

$$F(x) := H(x) - x \qquad (2.1).$$

And applies this operation as F(x) + x. This residual operation is implemented using "shortcut connections." Shortcut connections work by bypassing one or more layers, and the outputs of these connections are added to the outputs of the stacked layers. An important point to note is that these shortcut connections do not introduce additional parameters or computational complexity. The study, with its obtained results, demonstrates that optimizing the residual operation is easier compared to directly optimizing the fundamental operation using traditional methods. Consequently, deep neural networks can gain greater depth and complexity while being trained more efficiently.

2.1.4. You Look Only Once (YOLO)

Redmon et al. offers a new approach to object detection with YOLO (You Only Look Once) (Redmon et al. 2016). While previous methods perform object detection using classifiers, YOLO treats this task as a regression problem. A single neural network predicts bounding boxes and class probabilities directly from images, leading to a fast and effective detection system. The base model of YOLO is capable of processing in real time, and a smaller version can run at much higher speeds, outperforming other sensing systems. YOLO is a significant advancement in the field of object detection because it combines separate components into a single network, offering huge advantages in both speed and precision. This approach is ideal for real-time applications because all processing is done on a single network and therefore fast results can be obtained. Additionally, scores obtained by multiplying class probabilities and confidence scores provide detailed information about both object presence and accuracy, which can lead to better classification results. This combined detection approach of YOLO has generated great interest in the field of object detection and has been used for many applications. This technique has made a huge impact, especially in areas such as autonomous driving, security cameras and object recognition. YOLO's network architecture is inspired by the popular GoogLeNet model for image classification. The network consists of 24 convolutional layers and 2 fully connected layers. Instead of the inception modules used by GoogLeNet, a simpler approach of 1×1 reduction layers followed by 3×3 convolution layers is used. Among YOLO's major achievements, its ability to perform object detection at real-time operating speeds is particularly notable. This feature is an indispensable requirement for many applications such as video analysis, autonomous driving, security monitoring and many more. YOLO can be used effectively on video streams and to detect fast-moving objects without the need to process each frame separately. Additionally, YOLO's ability to learn representations of objects without being tied to a specific dataset is also a major achievement. This provides the flexibility to achieve successful results in different areas, from natural images to artistic works. This shows that YOLO is a versatile object detection solution and can be used for a wide range of applications. It is also important that YOLO has a background design that reduces the possibility of false positive predictions. This increases the reliability of detection results and makes systems less likely to generate unnecessary false alarms. This is a critical feature that increases the reliability of object detection applications. Finally, YOLO's ability to perform the entire object detection process within a single network provides the advantage of end-to-end training and optimization. This allows the network to produce more consistent and efficient results and makes the object detection process smoother. In this way, it contributes to making YOLO a more powerful and useful solution in practical applications.

2.2. Explainability Models

In this section, studies on Grad-CAM and Gradient Inputs models, which are among the XAI models we used to examine the black box of the deep learning models trained in our study, and the content of these models are examined.

2.2.1. Grad-CAM

In 2017, R. Selvaraju et al. introduced a technique known as Grad-CAM (Gradient-weighted Class Activation Mapping) to generate 'visual explanations' from CNN-based models for a wide range of classes (Selvaraju et al. 2017). This innovative method is designed to produce extensive localization maps that emphasize significant regions by utilizing the gradients flowing into the final convolutional layer, thereby offering deeper insights into the model's decision-making processes. Grad-CAM operates by leveraging the gradients from the last convolutional layer of the CNN. These gradients originate from the activations in the feature maps that influence the class score (the pre-softmax value) for a specified target class. The method involves several steps. First, the gradients for the target class are computed concerning the activations in the last convolutional layer. These gradients are then reduced in size using global average pooling to calculate the importance values for each neuron, indicating how crucial each feature map is for the target class. A linear combination of the feature maps, weighted by their importance values, is performed next. The resulting values are passed through the ReLU (Rectified Linear Unit) function to ensure that only features positively impacting the class of interest are highlighted, as negative values may pertain to other classes and are thus not emphasized. This process results in a heatmap that highlights the regions of the input image most relevant to the prediction for the target class, providing a visual representation of the model's focus areas and enhancing the interpretability of its decisions.

2.2.2. Gradient Inputs

The term "gradient inputs" is a technique used in explainability analyzes in deep learning models. This technique uses gradients (derivatives) to understand the effects of a data sample on the model output and how a particular input affects the model's prediction (Springenberg et al. 2014). Gradient Inputs is useful for quantitatively measuring the model's response to a specific input and the contribution of this response to each component of the input. This technique is used to better understand the internal structure of the model and decision processes. In particular, it is used to understand the effects of a particular input on changes in the model output. Gradient Inputs is usually calculated with the help of gradients. The gradients between the output of the model and the input reflect the effect of a particular input on the output. These gradients can be calculated separately for each component of the input and thus understand which features or components affect the output of the model more. The mathematical expression of Gradient Inputs contains derivatives of the model that describe the relationship between the input data and the model's outputs. Specifically, gradient values are calculated that measure the impact of each feature of an input on the output of the model.

2.3. Explainablitiy Metrics

In this section, faithfulness, maximum sensitivity and complexity metrics, which allow us to examine and compare how successfully the explainability of deep learning models trained for estrus detection are achieved, are examined.

2.3.1. Faithfulness

Faithfulness, in the context of XAI, refers to how accurately the descriptions of artificial intelligence models or algorithms reflect actual system behavior (Bhatt et al. 2020). Faithfulness is a concept that measures whether explanation methods or models help users or reviewers understand the model by accurately reflecting the true reasons and operation of the model's decisions. Faithfulness is used to evaluate whether explanatory methods or models help understand the internal working logic of the model and which features affect the outcome and how. If an explanation method or model is high in fidelity, it can better explain the model's decisions and help users have more faith in the reliability of the model. Because faithfulness is a crucial concept in assessing the capability of explanatory models or methods to accurately represent the functioning and decisions of the model in real-world applications. Faithfulness measures the accuracy of the relationship between an explanatory function g and an estimator (model) f. Where x represents the input data, xs refers to a particular subset of it and represents the change in the output f when we set the properties of this subset to the reference basis x̄s. Faithfulness can be expressed mathematically as follows;

$$\mu F(f, g; x) = corr(\{\sum i \in S \; g(f, x)i\} \; S \in P([d], \mid S \mid) \;, f(x) - f(x[xs = \bar{x} \; s])). \tag{2.2}$$

In this equation; μF(f,g;x) represents the faithfulness of annotation g to the estimator f.S represents the set of |S|-dimensional subsets containing all d elements. P([d],|S|) denotes the set of all these subsets. g(f,x) represents the reference value of the explanation process g to feature i in the estimator f. f(x) represents the prediction made by the estimator f for the input data x. f(x[xs= x̄s]) represents the change in the output of the estimator when the features in the subset xs are set to the reference baseline x̄s.

2.3.2. Maximum Sensitivty

Maximum Sensitivity measures how sensitive a model or annotation process is to the smallest changes in input data. That is, it answers the question of what effect a small change in the input data has on the model's output or explanation. This is used to evaluate the reliability and accuracy of explanations and models. If a model or explanation process is too sensitive to small changes in the input data, the reliability of that model or explanation decreases. Maximum Sensitivity helps us evaluate how reliable a model's output is. The model may be overly sensitive to some features, which may cause the model to introduce bias. Maximum Sensitivity can be used to detect such behavior of the model. A neighborhood set Nr of data points giving similar prediction results within a given radius r is created around a point x. For each data point z within this neighborhood, a metric D is used that measures the difference between the explanations produced by the explanation function g. Maximum sensitivity selects the z point that maximizes this metric and represents the sensitivity of the annotation at that point (Yeh et al. 2019);

$$\backslash muM(f,g,r;x) = maxz \in Nr\ D\big(g(f,x),g(f,z)\big). \tag{2.3}$$

2.3.3. Complexity

Describes the complexity of feature contribution annotations in the context of a predictive function f, annotation function g, and a data point x. The goal is to measure how simple or complex g is while providing an explanation of what properties of x are important in predicting the output of f. A higher entropy value indicates a more complex explanation, that is, the importance of different features is more unevenly distributed. Conversely, a lower entropy value indicates that the explanation is simpler and indicates that the contribution is concentrated on one or a few features. The main purpose of this complexity measure is to find a balance between compatibility with the model and understandability of the description so that users can better understand the description (Bhatt et al. 2020).

Given a predictor f, explanation function g, and a point x, the complexity of g at x is;

$$\mu C(f,g;\ x) =\ Ei\ -\ ln(Pg) =\ -X\ d\ i = 1\ Pg(i)ln\big(Pg(i)\big). \tag{2.4}$$

CHAPTER 3

MATERIALS AND METHODS

Within the scope of this study, the methods and the flowchart were applied as described in Figure 2. The flow chart includes the following steps; the dataset preparation and processing step was carried out. This stage includes collecting, cleaning and processing the dataset to be used. The dataset contains input data for training deep learning models. Deep learning models were trained on the created datasets. These models are neural networks optimized to perform a specific task. Grad-CAM and Gradient Inputs explainability models were created to evaluate the explainability of the trained deep learning models. These models were used to make the decisions and learning process of the deep learning model more understandable. To evaluate the success of the created explainability models, faithfulness, maximum sensitivity and complexity metrics were used. These metrics provide information on the faithfulness, maximum sensitivity, and complexity of explainability models.

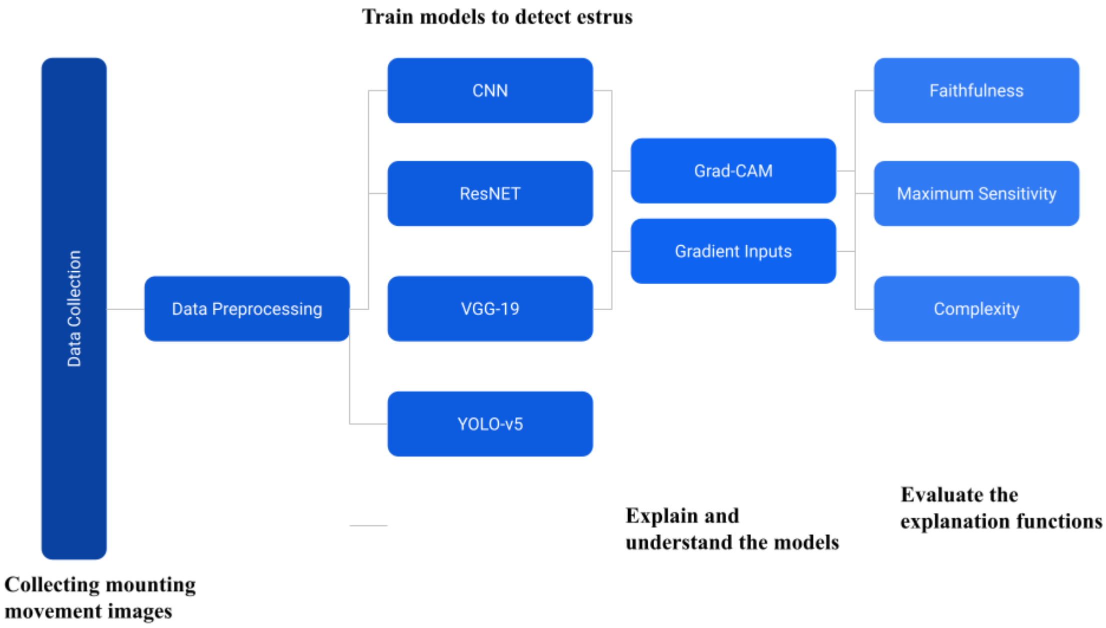

Figure 2. General Method.

3.1. Dataset

After the first calving, a dairy cow moves into the productive phase. During this period, the life cycle is listed as lactation, dry period and birth(Dallago et al. 2021). However, the difficulty of obtaining comprehensive data from different animals necessitated a video tagging process that took a year and required continuous observation. In order to reduce these difficulties, speed up the process and expand the dataset, a decision was made to use images from the Internet containing mounting or not-mounting behaviors. The research is centered around a dataset focusing on a variety of cows including Simmental, Holstein, Jersey and Montofon breeds.

Figure 3. Dataset positive class image collage (Arıkan et al. 2023).

This dataset contains images of cows exhibiting mounting behavior taken from different angles and images of cows not participating in such activities as seen in Figure 3 and Figure 4. The dataset was assembled by gathering images from online sources, particularly through search engines where cow images are publicly accessible. Each image was then manually tagged to ensure proper classification.

Figure 4. Dataset negative class image collage (Arıkan et al. 2023).

There are a total of 819 images containing two classes in the original dataset created. To prevent model overfitting, the dataset was enhanced using data augmentation techniques. During this phase, images were processed using methods such as rotation and zoom (Li et al. 2021). The total dataset size is 1638, of which 492 (30%) are designated as testing data and 1146 (70%) are designated as training data. The distribution is non-strategic and the two-class dataset contains images of 937 cows in estrus and 701 images of non-estrus cows, out of a total of 1638 images. The images were preprocessed and normalized before starting training; in this process, images with different pixel sizes were resized to (224,224) pixel sizes.

3.2. Deep Learning Methods for Estrus Detection

In this scope of our study, we conducted training sessions on our dataset using four distinct models developed for estrus detection in cows through image analysis. These models include the CNN, VGG-19, ResNet, and YOLO-V5. Each model is specifically tailored for the task of identifying estrus patterns in cow images. The CNN model is designed to extract intricate features from the images, enhancing its capability to discern subtle details indicative of estrus behavior. The VGG-19 model, leveraging transfer learning, exploits pre-existing knowledge to augment its proficiency in recognizing estrus-related patterns in cow images. Additionally, the ResNet model, with its deep residual learning architecture, contributes to the learning of complex patterns essential for accurate estrus detection. The YOLO-V5 model, known for its real-time object detection prowess, is employed to efficiently locate and classify estrus-related features in a single comprehensive step. By training these diverse models on our dataset, we aim to compare their performance and evaluate their suitability for the specific task of estrus detection in cows.

3.2.1. Convolutional Neural Network

ANN mimics the functioning of the human brain to facilitate learning, interpret acquired information, and make autonomous decisions. CNN, a specialized type of ANN, are designed for image recognition and computer vision tasks. They enable computers to interpret incoming images by converting them into a matrix format that is computationally manageable. The first layer in a CNN is the convolution layer, which applies various filters to the input image to extract features like edges and corners. The output from this layer is then passed through an activation function to introduce nonlinearity into the model. Following this, the output is sent to a pooling layer, which reduces the size of the feature maps while retaining essential information. Typically, the final output of a CNN is generated by a fully connected layer, which performs classification. This layer learns the significance of differences during the training phase and utilizes this knowledge to make predictions on new images.

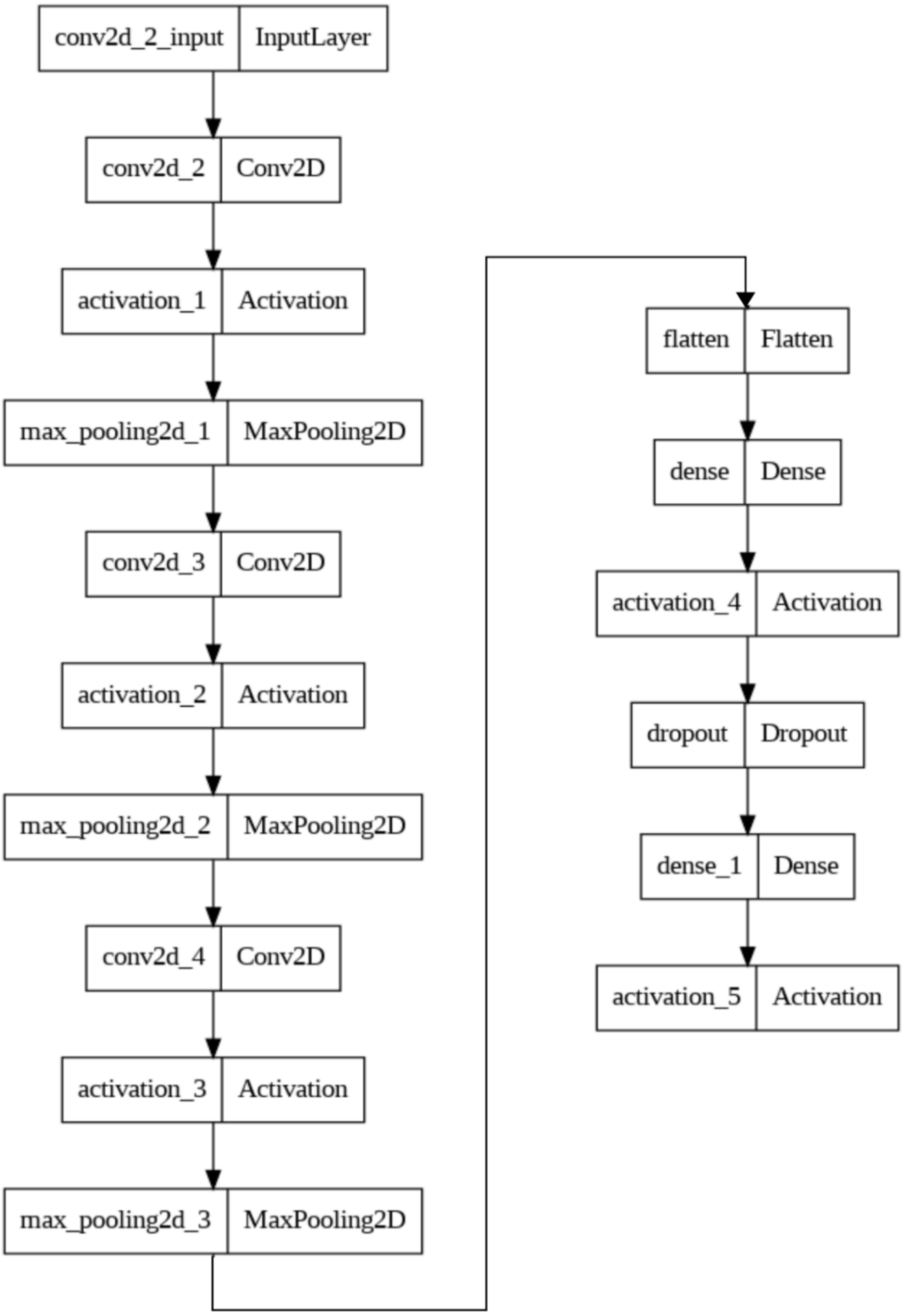

Figure 5. CNN model's architecture.

The CNN architecture designed for estrus detection in cows consists of 9 layers in total as seen in Figure 5. The model structure represents a CNN with 9 layers, including 3 convolution layers that learn hierarchical features, 3 pooling layers that reduce spatial dimensions, 1 flattening layer that enables the transition from convolution/pooling layers to fully connected layers, and 1 densely connected layer for general pattern recognition. The model contains 1 dropout layer to prevent overfitting and 1 output layer for binary classification. This particular model is designed to distinguish whether a cow is in estrus period or not. The use of the “binary_crossentropy” loss function and the “rmsprop” optimizer during compilation reflects the nature of the binary classification task. The "Accuracy" metric is used to evaluate the performance of the model, providing a clear measure of its ability to accurately classify into the desired binary categories. These comprehensive architecture and compilation settings are intended to optimize the model's accuracy in predicting estrus detection results.

3.2.2. VGG-19

In the context of this research, the VGG-19 model underwent training on a dataset comprising two distinct classes. This dataset encompasses images belonging to these predetermined classes and has been appropriately divided for both training and testing purposes. The VGG-19 architecture, a custom CNN developed by the Visual Geometry Group (VGG), was selected for this study.

Recognized as a 19-layer deep CNN specifically designed for image classification tasks, the VGG-19 architecture features convolutional layers with 3×3 filters, followed by max pooling layers and rectified linear unit (ReLU) activation functions. The final layers of the VGG-19 model are fully connected layers that handle the classification task, as illustrated in Figure 6. The output from these layers is passed to a SoftMax activation function to produce class probabilities. In this study, the fully connected layer of the pre-trained VGG-19 model was removed and replaced with a new connection layer tailored to the number of classes in the dataset. By leaving the upper layers of the model fixed, they are prevented from being updated during training. Then, a new connection layer is added on top of the model to perform the two-class classification task. The model is retrained on the training dataset; in this process, 'Rmsprop' optimization algorithm is used to minimize the determined loss and training is performed for 20 epochs.

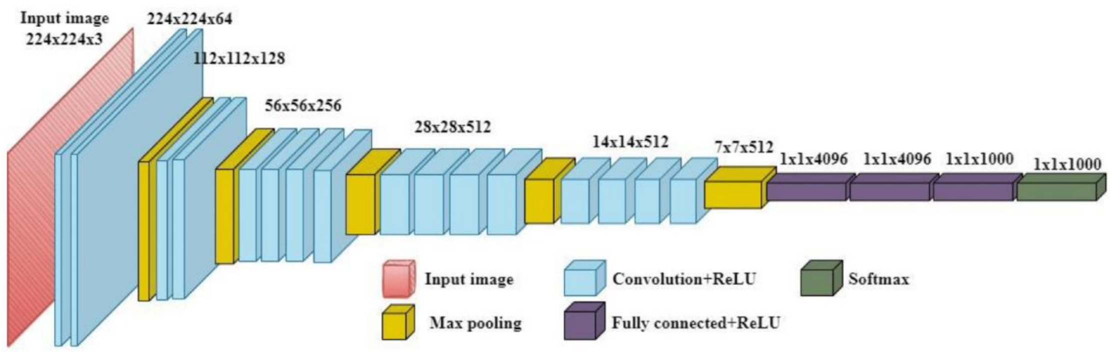

Figure 6. VGG-19 model's architecture.

3.2.3. ResNet

ResNet architecture includes a complex structure of redundant blocks to facilitate learning complex models. ResNet is a deep convolutional neural network characterized by skip connections that ensure smooth gradient flow during training and reduce the vanishing gradient problem. The model is structured with multiple residual blocks containing convolutional layers, normalization, and activation functions as described in Figure 7. This architecture facilitates effective capture of hierarchical features critical for estrus detection. Pooling layers help reduce spatial dimensions, and the model is further enhanced with a flattened layer for seamless transition to fully connected layers. A single fully connected layer is used for global pattern recognition. The final output layer is configured for binary classification, which determines the estrus status of cows. The model was compiled with the "binary_crossentropy" loss function and the "rmsprop" optimizer in accordance with the binary nature of the classification task. Evaluation of the ResNet model is performed using the "accuracy" metric, which provides a reliable measure of its performance in correctly classifying cows. Designed for this specific application, the ResNet architecture aims to improve learning capabilities and overall prediction accuracy with its unique residual connections.

To train a two-class transfer learning model for the ResNet model, first the upper layers of the model are left constant, preventing these layers from being updated during training. Then, a new connection layer is added on top of the model to perform the two-class classification task. The model is retrained on the training dataset; In this process, 'Rmsprop' optimization algorithm is used to minimize the determined loss and training is performed for 30 epochs.

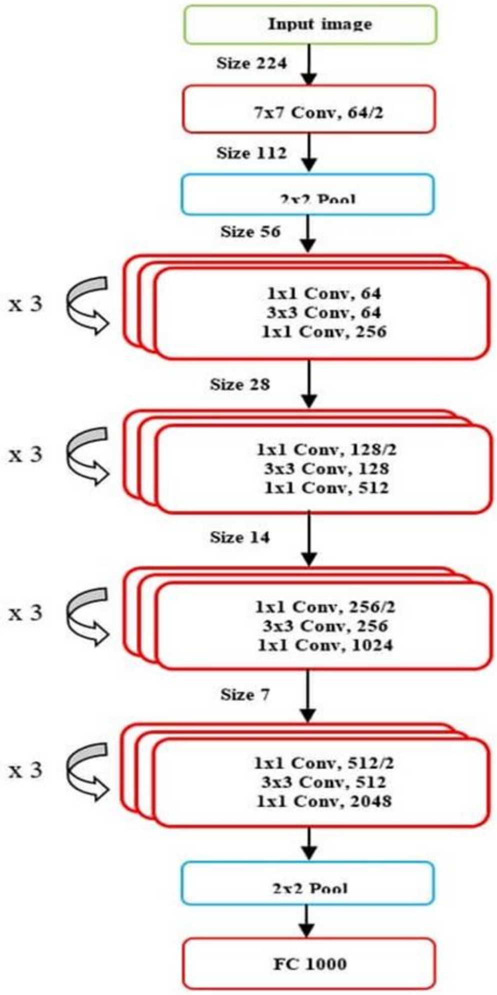

Figure 7. ResNet model's architecture.

3.2.4. YOLO

YOLOv5 is a deep learning model used in the field of object detection. The basis of the architecture is a network called CSPDarknet53, which provides a structure similar to Darknet53 in previous YOLO versions. YOLOv5 integrates information at different scales using PANet (Path Aggregation Network) to combine feature maps as described in Figure 8. The header section determines the class of the object and predicts its coordinates. The model usually includes common activation functions such as ReLU and a total loss function combining Focal Loss and regression loss for the object detection task. When using SGD or Adam optimization algorithms in training, data augmentation techniques increase the generalization ability of the model. YOLOv5 offers an architecture that offers faster and more effective object detection compared to previous versions; this provides a powerful solution for real-time applications and general object detection tasks. In our study, the dataset was manually labeled to distinguish between positive and negative classes. We trained the YOLOv5 model, specifically version 5 of YOLO, for 150 epochs using this labeled dataset.

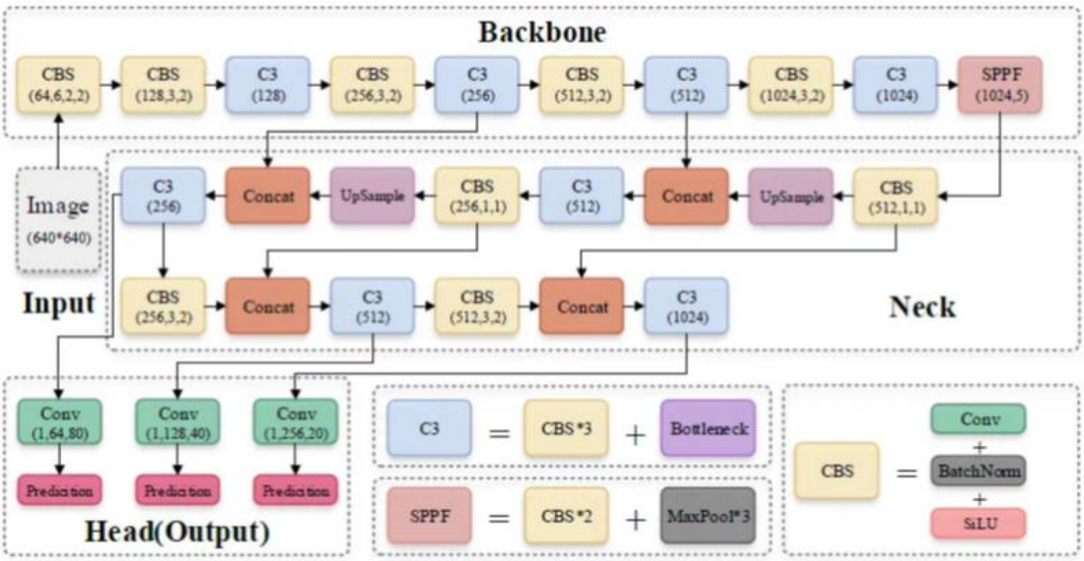

Figure 8. YOLO model's architecture.

3.3. Explanation Functions for Deep Learning Models

In this section, we will study on the interpretability of deep learning models specifically developed for estrus detection. Our emphasis is on scrutinizing the features that our detection models prioritize during decision-making, utilizing both the Grad-

CAM and Gradient Inputs methods. These techniques facilitate a comprehensive understanding of the specific features within the images that significantly influence the model's decision-making process.

Grad-CAM proves valuable when seeking insights into how the model perceives objects or features, providing visualizations of object locations. It becomes especially handy when attempting to discern which sections the model emphasizes for class prediction. On the other hand, Gradient Inputs is beneficial for comprehending how the model responds to the image and determining which pixels contribute more to feature detection. This method is particularly useful when aiming to understand specific features within the image and identifying the input features that capture the model's focus.

As a result, Grad-CAM and Gradient Inputs serve distinct analytical purposes, and the choice between them depends on the specific objectives of the analysis. Grad-CAM, primarily employed for visual tasks, generates a heatmap highlighting image regions influencing the model's decision. Conversely, Gradient Inputs analyze the contribution of input features by calculating gradients of the model's output on these features. Grad-CAM offers a visually interpretable technique well-suited for image classification tasks. On the contrary, Gradient Inputs, being a more versatile method, can be applied to various model architectures and data types. While Grad-CAM provides interpretable visual descriptions, Gradient Inputs offer a more direct yet visually straightforward analysis. Utilizing both approaches enables a more comprehensive understanding of our model and enhances the detection of relevant features.

3.3.1. Grad-CAM

In the Grad-CAM study, which was conducted to evaluate the features that the developed models prioritize in the decision-making process, it becomes clear which features the models emphasize when classifying. Features highlighted in red were identified in the predictions of the developed CNN, ResNet and VGG-19 models.

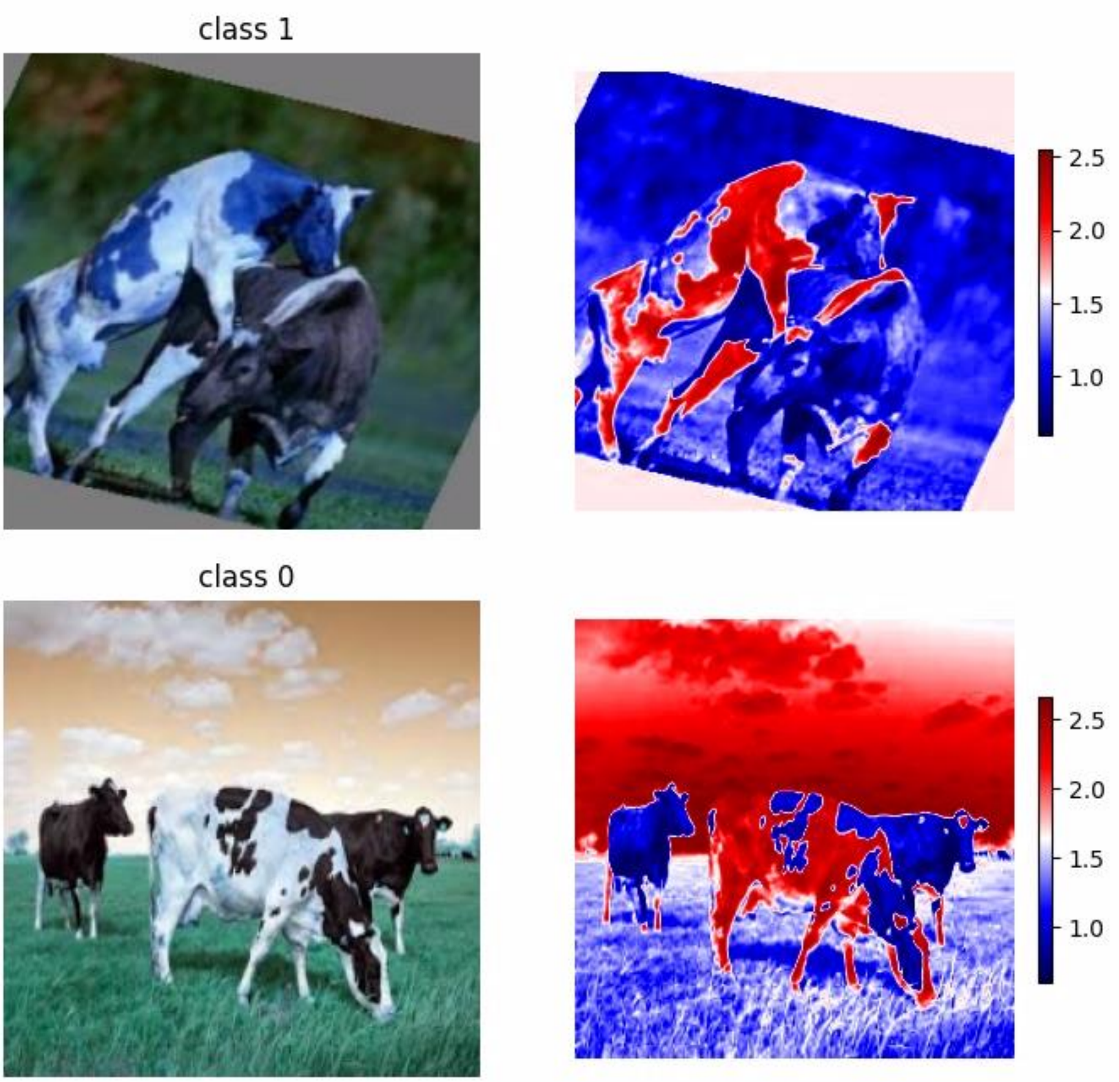

Figure 9. CNN model's Grad-CAM explainability examples.

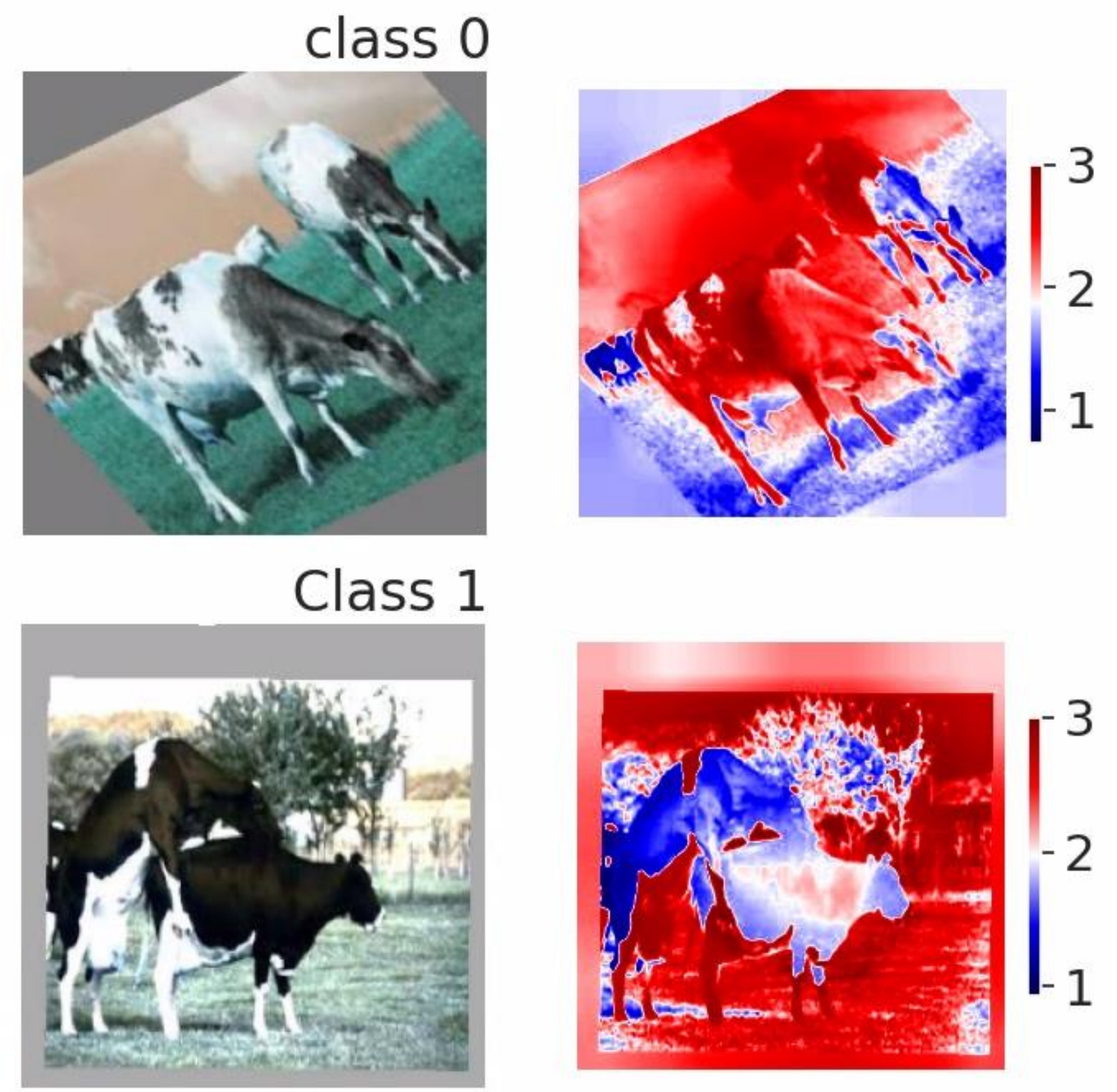

Figure 10. ResNet model's Grad-CAM explainability examples.

In particular, in all three models, the dorsal and udder regions of the cows emerge as the features most prominently used to distinguish positive classes. These findings indicate that the models specifically focus on these anatomical regions in the classification process and point to the important role of these regions in determining positive classes as seen in Figure 9, Figure 10 and Figure 11.

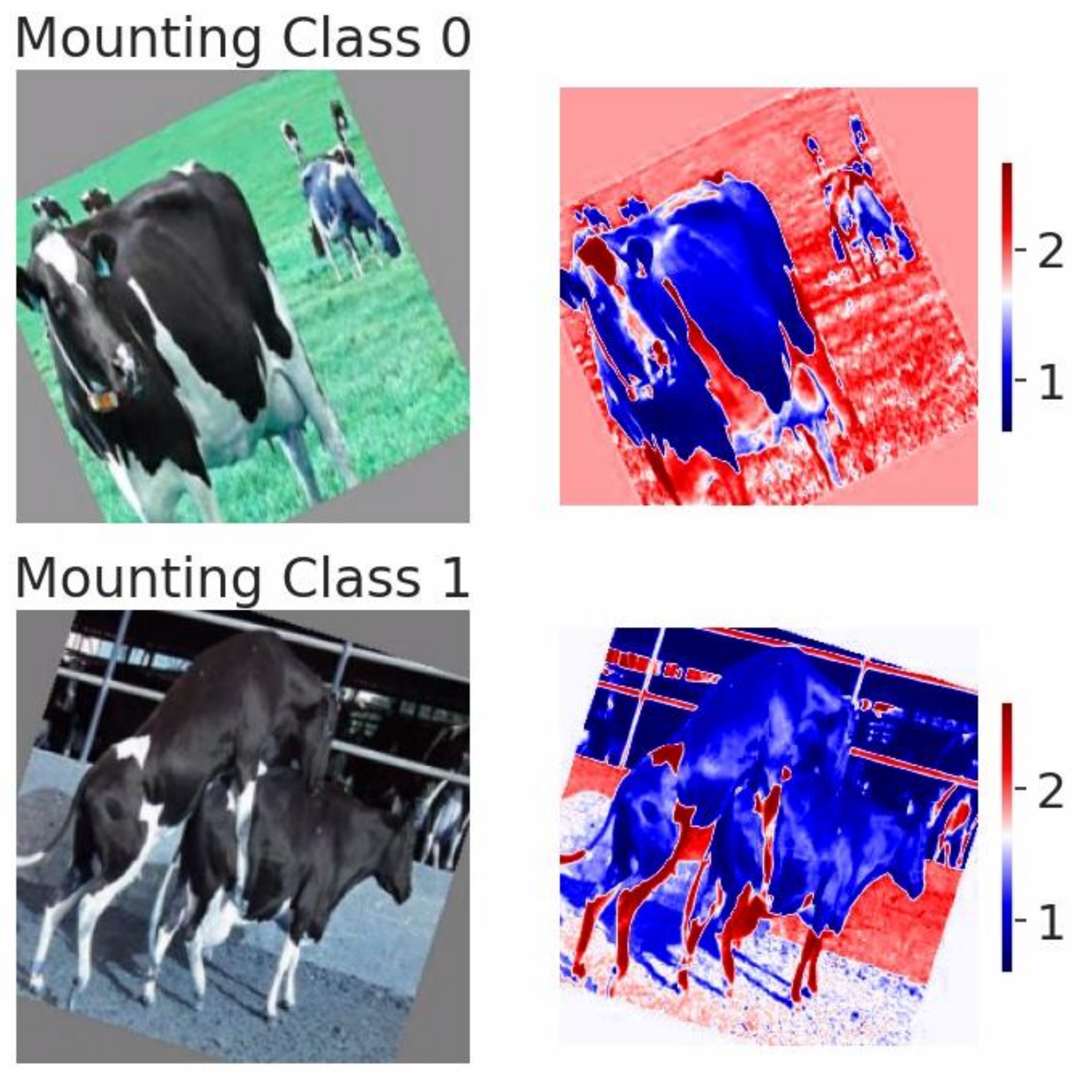

Figure 11. VGG -19 model's Grad-CAM explainability examples.

3.3.2. Gradient Inputs

In this research, another deep learning model method used for explainability in this research is Gradient Inputs. Mathematically, the Gradient Inputs method includes the following steps: First, an input sample is passed through the model and the output is obtained. Gradients are then calculated with derivatives based on the inputs. These gradients represent the sensitivity of the output to the input. Dot product is performed between input features and gradients. This process determines how effective the input features are by multiplying them by gradients.

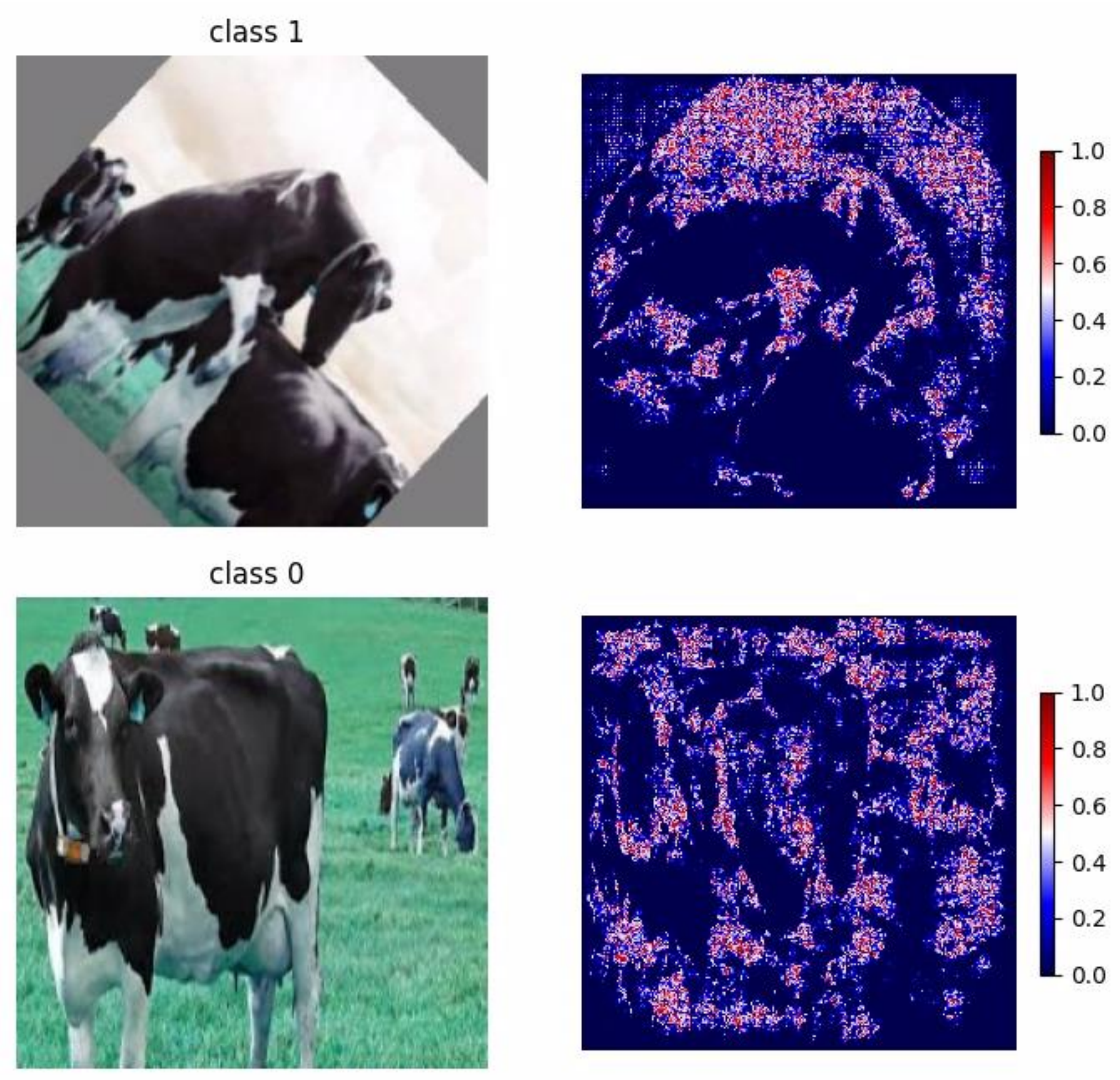

Figure 12. CNN model’s Gradient Inputs explainability examples.

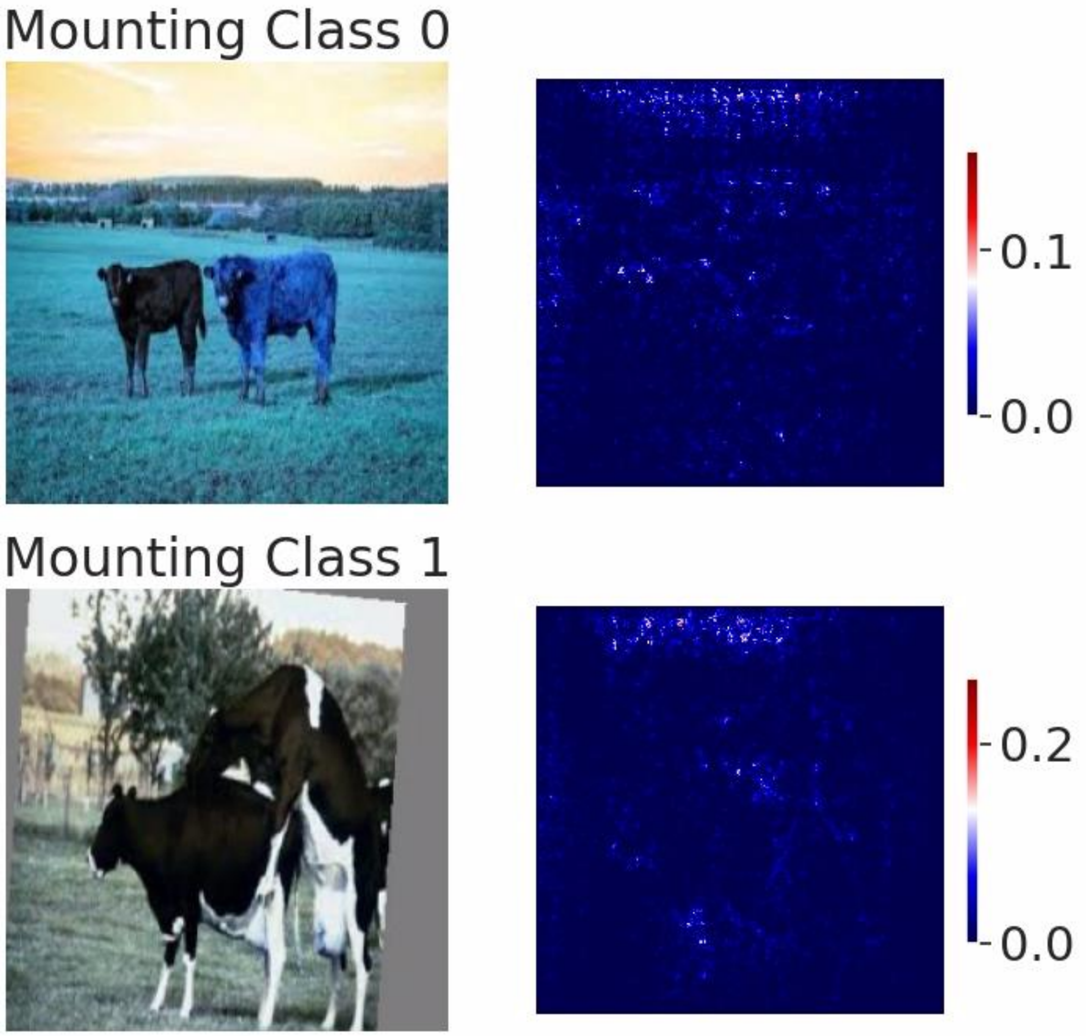

Figure 13. ResNet model’s Gradient Inputs explainability examples.

The resulting product creates an importance map that represents the impact of input features on the output. This heatmap reveals how influential input features are in determining the output. In the Gradient Inputs study, which was carried out to evaluate which features the developed models focus on in the decision-making process, it can be clearly observed which features the models give priority to when classifying. In all three models, the most used feature to distinguish positive classes is the hindquarters of the cows as seen in Figure 12, Figure 13 and Figure 14.

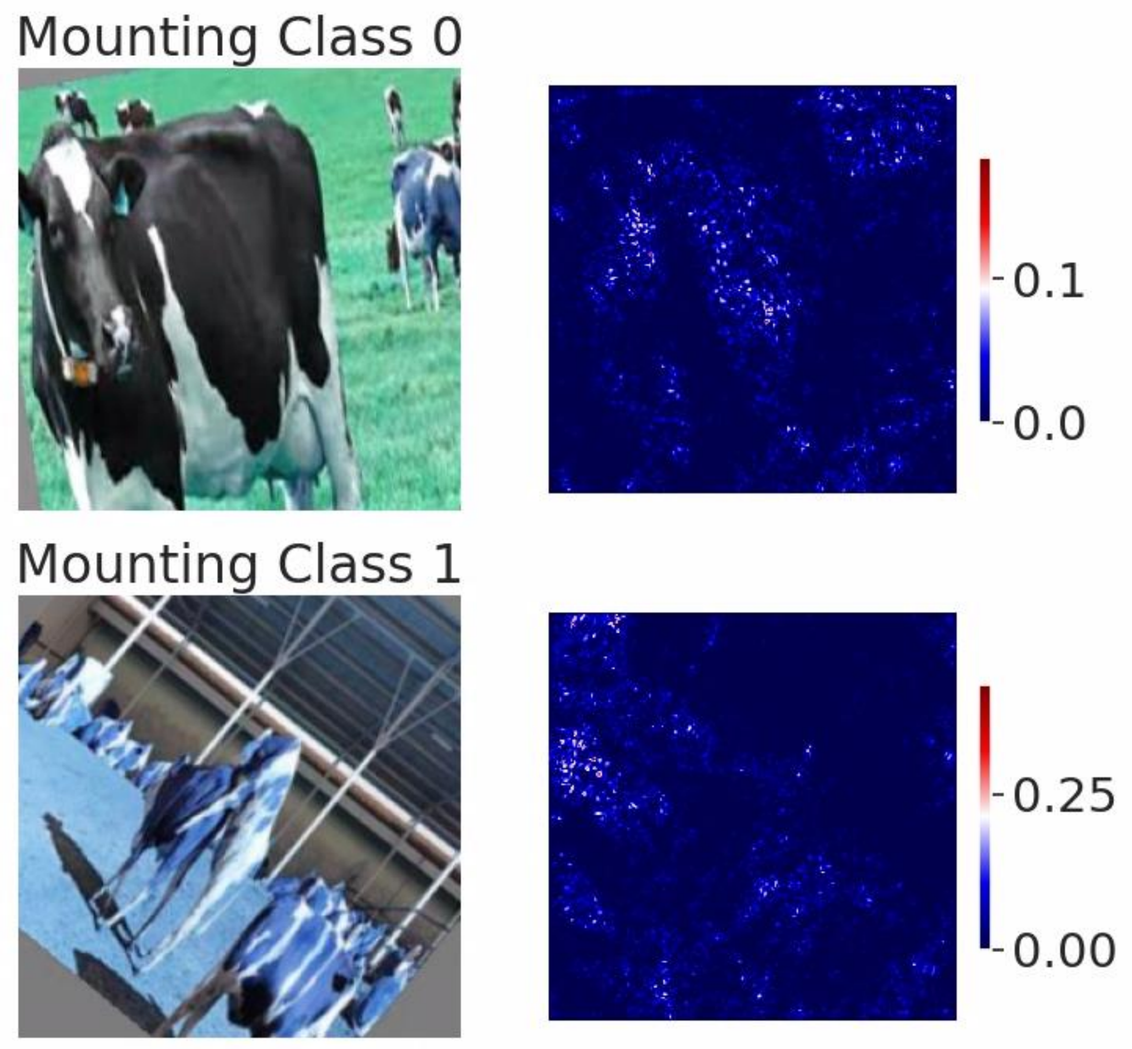

Figure 14. VGG-19 model's Gradient Inputs explainability examples.

3.4. Evaluating Explaination Function

In this part, we have explored the efficacy of interpretability techniques in explaining our models. To assess the efficiency of these techniques, we have computed three distinct metrics. In addition to the faithfulness metric used for gradient-based methods, we have also computed maximum sensitivity and complexity metrics.

3.4.1. Faithfulness

Faithfulness refers to how accurately an annotation method represents important features that contribute to a model's output. The Faithfulness metric measures the correlation between the sum of allocation scores of a selected subset of features and the difference in the model's output when those features are set to a reference baseline. The reference baseline can be determined in different ways, such as a value close to zero for the model's output or the average of the training data. Mathematically, the faithfulness measure is calculated as follows: Given a prediction model (f), an explanation function (g), an input (x) and a subset size (|S|), the faithfulness measure calculates the following correlation:

$$\mu F(f, g;\ x)\ =\ corr\ S \in (\ [d]\ |S|) \tag{3.1}$$

In this formula, |S| where xs represents the size of the subset, xs represents a subvector, and x [xs=x¯s] represents an input where xs features are set to a reference baseline.

In this study, the faithfulness metric was calculated for all explainability models applied for three deep learning models: CNN, ResNet and VGG-19.

3.4.2. Maximum Sensitivity

Maximum sensitivity refers to the highest sensitivity of an annotation method around a given point. This is calculated by comparing the outputs of the annotation function with other points in the point of interest. Mathematically, maximum sensitivity (μM) is calculated using the prediction model (f), description function (g), radius (r), and point of interest (x) as follows:

$$\mu M(f, g, r;\ x) =\ max\ z\ \in Nr\ D\big(g(f, x), g(f, z)\big). \tag{3.2}$$

High maximum sensitivity indicates significant differences between annotation function outputs at different points around the point of interest. Low maximum sensitivity indicates that the annotation function outputs at different points around the point of interest are more consistent and similar. Explainability of deep learning models is important for understanding the internal structure of the model and making reliable decisions.

In this study, the maximum sensitivity metric was calculated for all explainability models applied for three deep learning models: CNN, ResNet and VGG-19.

3.4.3. Complexity

It accounts for the complexity of feature contribution descriptions in the context of a prediction function f, an explanation function g, and a data point x. The goal is to measure how simple or complex g is when describing which features are important in predicting the outcome of x. A higher entropy value indicates a more complex description, that is, the importance of different features is more unequally distributed. A lower entropy value indicates a simpler explanation and a concentration of the contribution on one or a few features. This complexity measure aims to find a balance between compatibility with the model and understandability of the explanation, so that users can better understand the explanation. Given an estimator f, description function g, and a point x, the complexity of g is mathematically:

$$\mu C(f, g; x) = Ei - ln(Pg) = -X\ d\ i = 1\ Pg(i)\ ln(Pg\ (I)). \tag{3.3}$$

In this study, the complexity metric was calculated for all explainability models applied for three deep learning models: CNN, ResNet and VGG-19.

CHAPTER 4

RESULTS

The dataset, which includes images capturing the mounting behavior of cows during the estrus period, was utilized to successfully detect estrus. Four different deep learning models; CNN, VGG-19, ResNet, and YOLO-v5 were trained on this dataset. The trained CNN, VGG-19, and ResNet models were analyzed using Grad-Cam and Gradient Inputs explainability methods. Subsequently, the explainability evaluation was conducted using faithfulness, maximum sensitivity, and complexity metrics.

Table 1. Estrus Detection model's hyperparameters and accuracies.

Model	Loss Function	Optimizer	Performance Metric	Train Data	Validation Data	Loss	Accuracy
CNN	Binary cross entropy	Rmsprop	Accuracy	%70	%30	0.02	%98.22
VGG-19	Binary cross entropy	Rmsprop	Accuracy	%70	%30	0.01	%99
ResNET	Binary cross entropy	Rmsprop	Accuracy	%70	%30	0.01	%99.18
YOLO-v5	Binary cross entropy	Rmsprop	Accuracy	%70	%30	0.02	%98

4.1. Deep Learning Models for Estrus Detection

4.1.1. Convolutional Neural Network

An effective CNN model was developed to tackle the classical image recognition challenge. Comprising nine layers, this CNN model was trained on a dataset divided into 70% for training and 30% for testing, as illustrated in Table 1. After 20 training epochs, the model attained an accuracy of 98% on the training dataset, with a corresponding loss value of 0.02, depicted in Figure 15. Proper evaluation of CNN outcomes entails grasping the metrics employed, taking into account both the dataset and training data, and conducting a thorough analysis of the results.

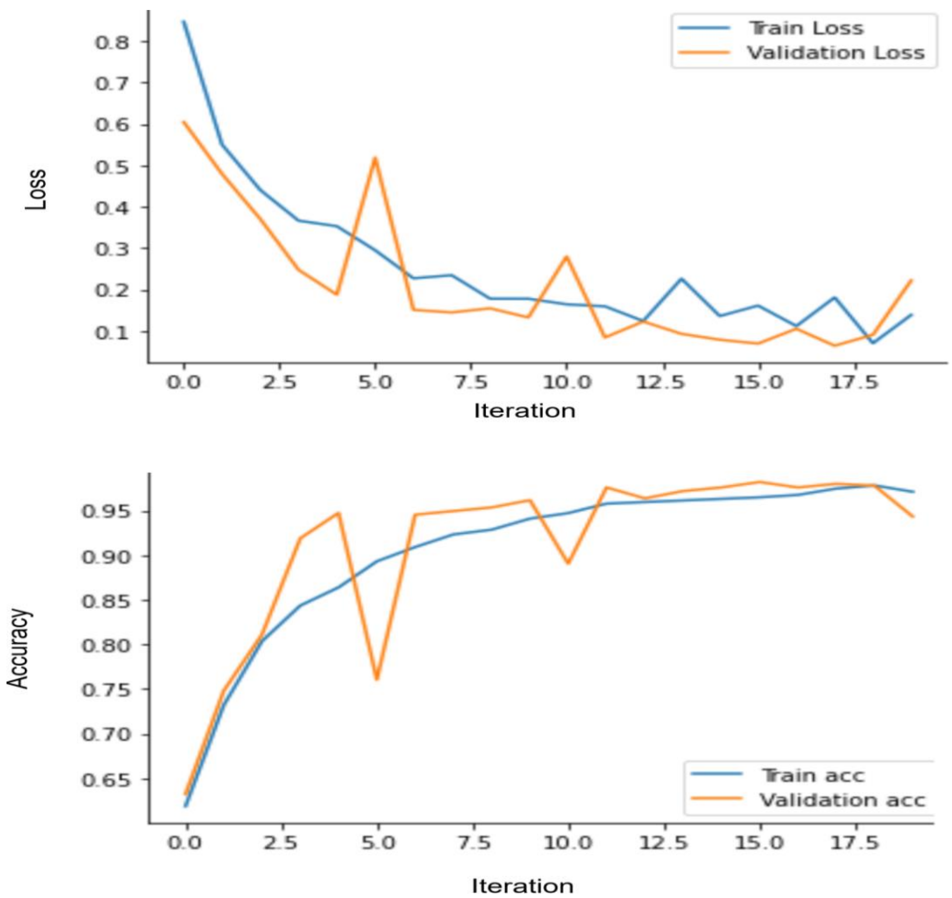

Figure 15. Loss and accuracy rates for CNN model to detecting estrus.

In the test dataset, the confusion matrix reveals that the CNN model accurately detected 203 out of 207 estrus cases and correctly identified 279 out of 285 estrus negative cases, as seen in Figure 16.

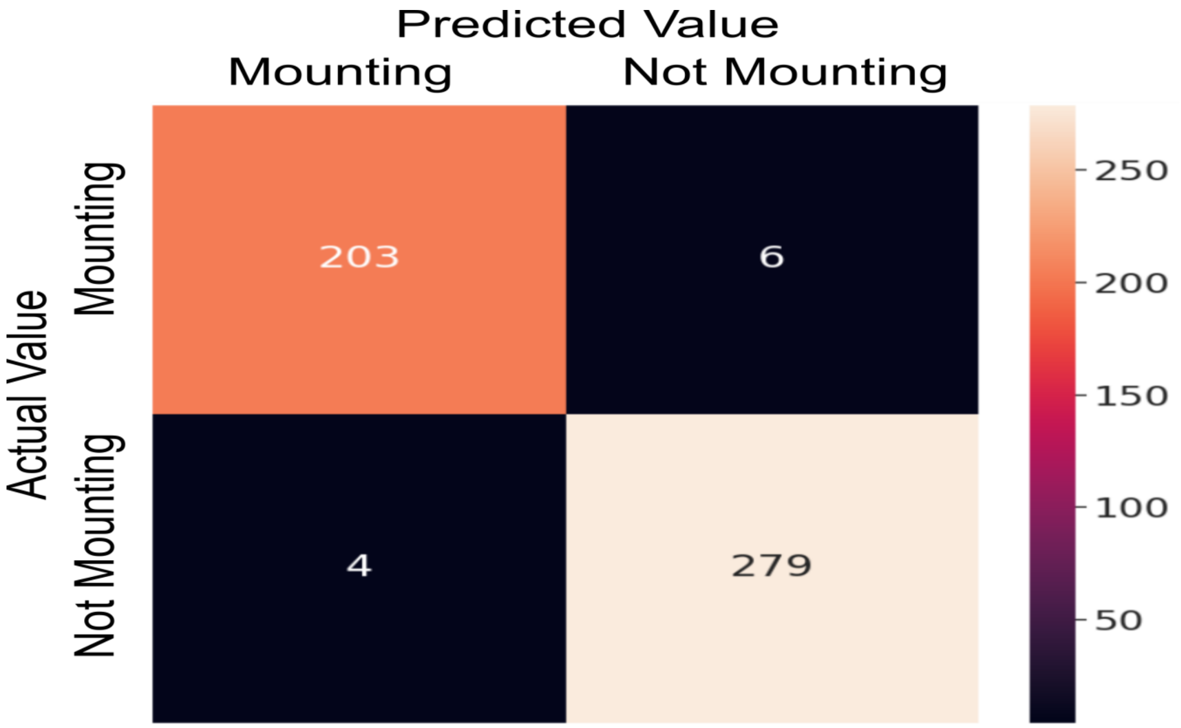

Figure 16. Confusion matrix for CNN model to detecting estrus.

4.1.2. VGG-19

In this study, the aim was to detect the estrus periods of cows by using VGG-19 as a transfer learning model. In training the model, the dataset was divided into 70% training and 30% testing, and a fully connected layer was arranged to focus on the two classes that determine estrus states as seen Table 1. The model was trained for 20 epochs and demonstrated successful performance, reaching a 99% accuracy rate as seen in Figure 17. The confusion matrix shows that it correctly predicted the estrus states in the test dataset. Additionally, the developed model correctly identified 280 out of 283 negative states, showing that it successfully predicted 209 estrus states in the test dataset as seen in Figure 18. However, it has been determined that the source of incorrect predictions is due to data augmentation techniques used to prevent overfitting. Rotation, zooming and other operations applied during data augmentation processes caused pixel losses, which caused the model to make incorrect classifications. In conclusion, although the VGG-19 model has achieved high accuracy in estrus detection, it is important to carefully manage data augmentation processes and take additional measures to prevent overfitting.

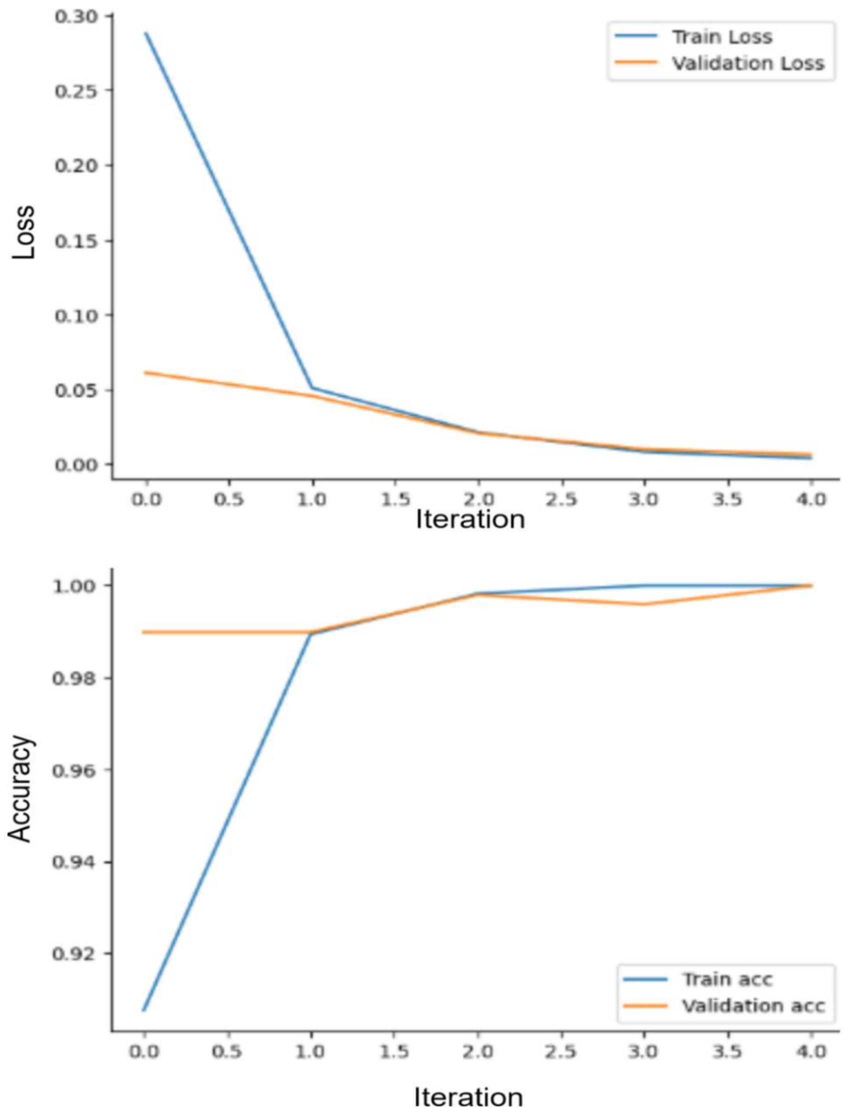

Figure 17. Loss and accuracy rates for VGG-19 model to detecting estrus.

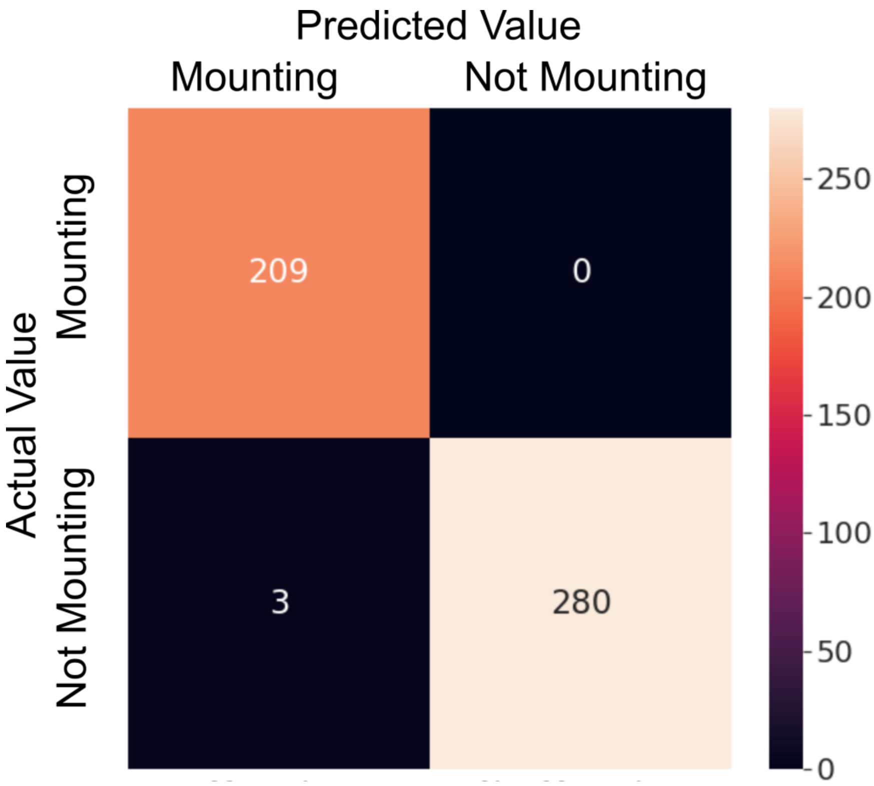

Figure 18. Confusion matrix for VGG-19 model to detecting estrus.

4.1.3. RESNET

In this study, the aim was to use ResNET as a transfer learning model to detect the estrus periods of cows. The dataset was split into 70% for training and 30% for testing, and a fully connected layer was configured to focus on the two classes determining estrus states as seen Table 1. The model was trained for 20 epochs and demonstrated successful performance, achieving a 99% accuracy rate as seen in Figure 19. The confusion matrix indicated accurate predictions of estrus states in the test dataset. Furthermore, the developed model accurately identified 281 out of 283 negative states, successfully predicting 208 estrus states in the test dataset as seen in Figure 20. However, upon examining the misclassification results, similar to the VGG-19 model, it was observed that the inaccuracies stemmed from augmented data.

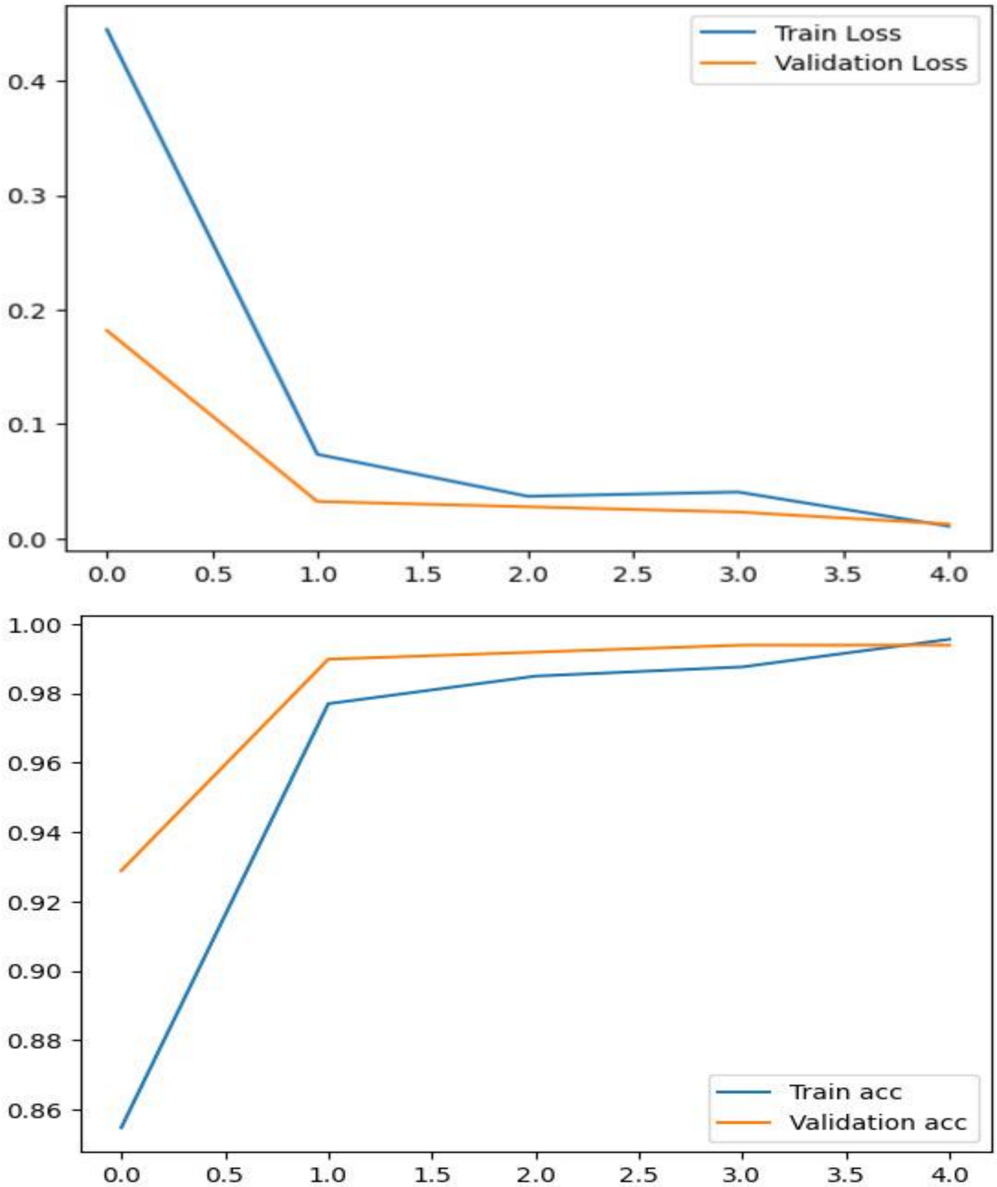

Figure 19. Loss and accuracy rates for ResNet model to detecting estrus.

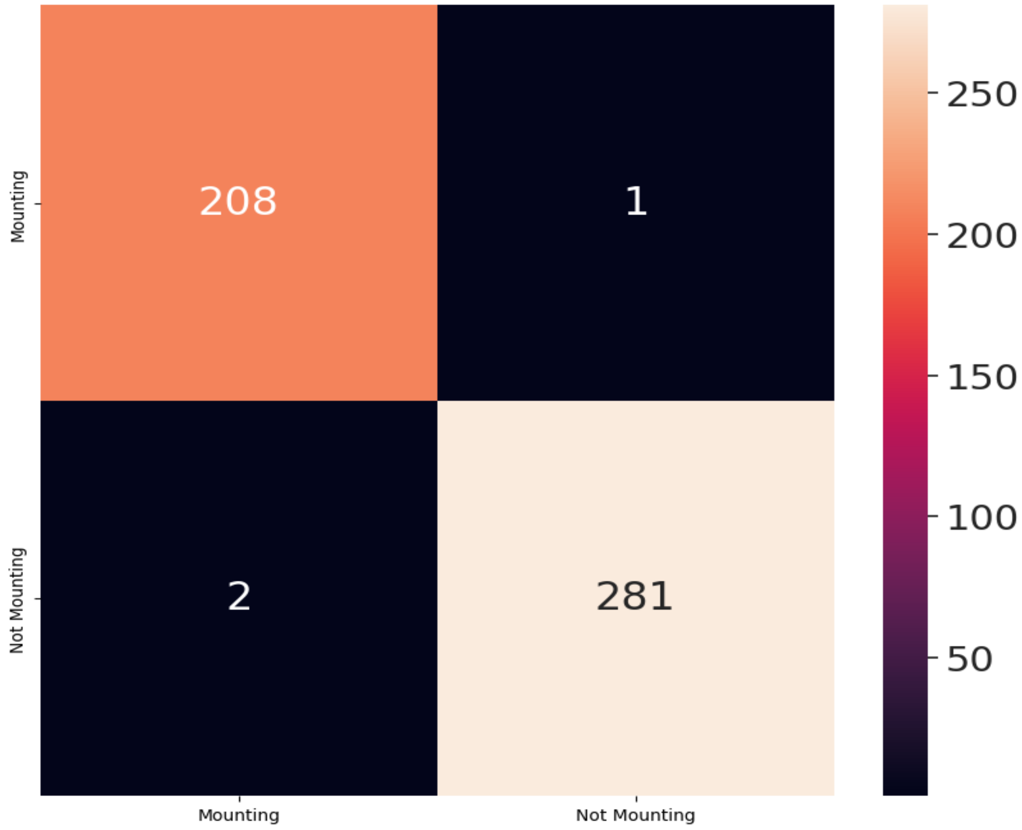

Figure 20. Confusion matrix for ResNet model to detecting estrus.

4.1.4. YOLO-v5

YOLO-v5 deep learning model was used on a dataset containing 937 images to detect estrus states. Various metrics were used to evaluate the performance of YOLOv5; common metrics used to evaluate object recognition models include average mean precision (mAP), accuracy, precision, and recall. mAP is particularly prevalent and serves as a comprehensive measure of the model's success in recognizing objects within an image. A high mAP indicates that the model is accurate and reliable. Accuracy measures the rate at which the model makes a correct prediction. The YOLOv5 model, once trained, achieved a 98% accuracy rate in detecting estrus states, as seen in Figure 21. The "metrics/mAP 0.5" value represents the accuracy rate of the objects successfully detected by the model. Additionally, the "Loss" values indicate the errors encountered by the model during training, with the model persisting in training until it minimized these loss values.

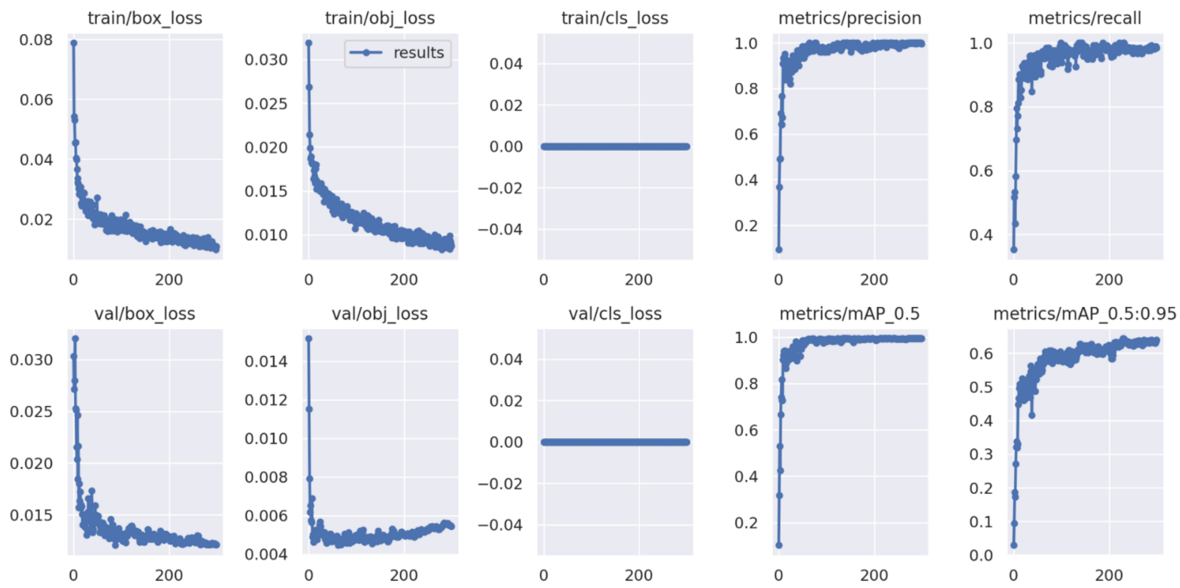

Figure 21. Loss and accuracy rates for YOLO-v5 model to detecting estrus.

4.2. Evaluations Results for Explainable Artificial Models

Faithfulness, maximum sensitivity, and complexity metrics are vital in interpreting explainability models. These metrics measure how well a model's predictions fit the input data, what features shape the model's decisions, and how complex the model is. Faithfulness measures how well a model's predictions fit the input data. This indicates how accurately the model reflects the real-world situation and plays a critical role in assessing reliability. Maximum sensitivity measures the model's ability to recognize the most important features in the decision process. This helps in understanding the behavior of the model by determining which features affect the model's results the most. Complexity metrics evaluate the complexity and understandability of the model. Less complex models can often be more easily interpreted and reliable, so complexity metrics are important to optimize the model's performance and increase its understandability. Together, these metrics help increase the accuracy, interpretability, and reliability of explainability models. Therefore, when working on explainability models, it is important to carefully examine and evaluate these metrics.

4.2.1. Faithfulness

In the field of deep learning models, explainability is important to understand the inner workings of the model and make reliable decisions. In this study, we evaluated the explainability models of Grad-CAM and Gradient Inputs for ResNet and VGG-19 models along with the CNN model. As seen in Figure 22; the Grad-CAM explainability model achieved a faithfulness value of 0.33, indicating its success in highlighting important features for class predictions in the CNN model. This indicates that Grad-CAM's CNN model is a reliable tool for understanding the decision process. For the same CNN model, the Gradient Inputs explainability model gave a faithfulness value of 0.26. This result suggests that the Gradient Inputs model may provide incomplete or inaccurate explanations in certain cases and may point to potential limitations. When we focused on the ResNet model, the faithfulness value of the Grad-CAM explainability model was calculated as 0.20. This low faithfulness rate reflects the difficulty of the Grad-CAM model in dealing with the complexity of the ResNet model, potentially leading to less reliable predictions than the CNN model. For the ResNet model, the Gradient Inputs explainability model gave a faithfulness value of 0.15. This result suggests that the Gradient Inputs model may provide incomplete or inaccurate explanations in certain cases and may point to potential limitations. When we focused on the VGG-19 model, the faithfulness value of the Grad-CAM explainability model was calculated as 0.28. This low faithfulness rate indicates the difficulty of the Grad-CAM model in dealing with the complexity of the VGG-19 model and, like the ResNet model, could potentially lead to less reliable predictions than the CNN model. For the VGG-19 model, the Gradient Inputs explainability model gave a faithfulness value of 0.13. As a result, when the results of Grad-CAM and Gradient Inputs explainability models are compared for both CNN and ResNet and VGG-19 models, it is seen that Grad-CAM offers higher faithfulness values in both cases. These findings highlight the importance of choosing the Grad-CAM method to increase the explainability of deep learning models, especially when dealing with models with complex architectures.

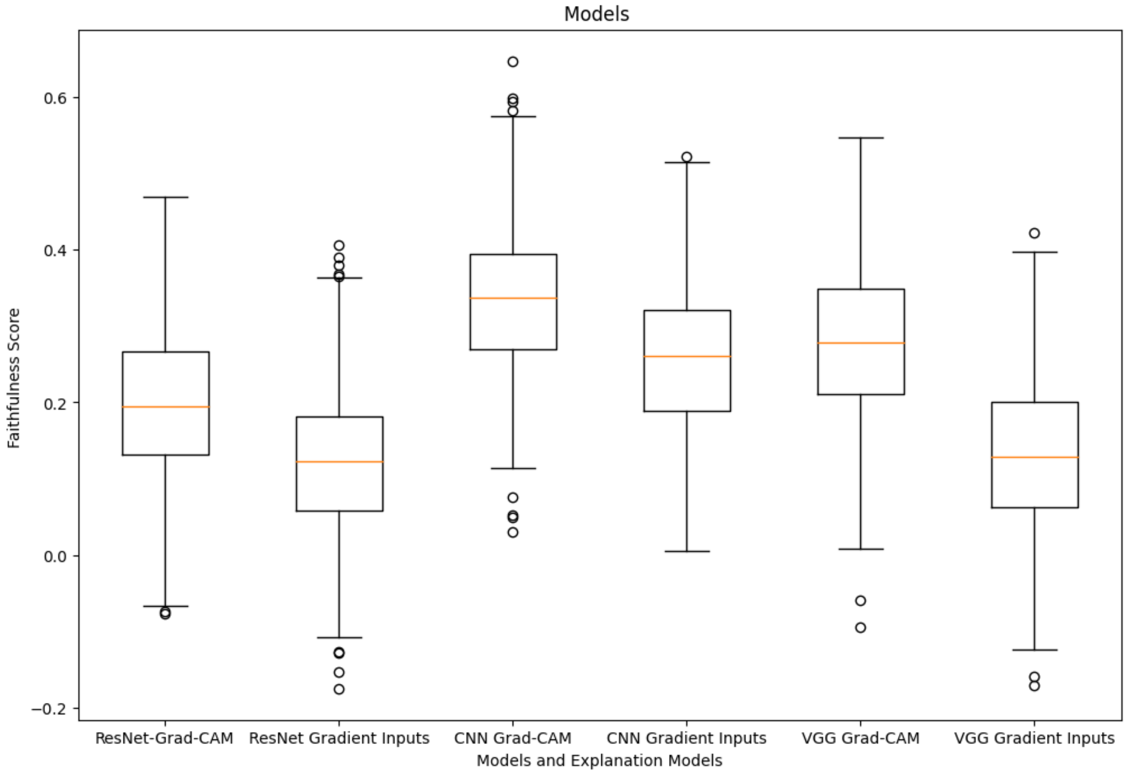

Figure 22. Faithfulness scores for deep learning models to detect estrus.

4.2.2. Maximum Sensitivity

In this study, we evaluated Grad-CAM and Gradient Inputs explainability models, focusing on maximum sensitivity values for both CNN and ResNet models. As seen in Figure 23; the maximum sensitivity value for the Grad-CAM model applied for CNN was calculated as 0.32. This value shows that the Grad-CAM model consistently highlights explanations around data points for CNN predictions. However, we note that given the relatively high value, there may be ambiguities and differences in the descriptions of certain data points. Similarly, the Gradient Inputs model for CNN yielded a maximum sensitivity value of 0.42, indicating possible ambiguities and differences in descriptions. Lower maximum sensitivity values mean more consistent and precise descriptions around data points. For the ResNet model, the Grad-CAM model showed a maximum sensitivity value of 0.15, indicating more consistent reflections of predictions with less variation in explanations. On the other hand, the maximum sensitivity value of the Gradient Inputs model for ResNet was 0.17; This suggests higher variance in explanations around data points, potentially leading to less reliable insights at certain points. For the VGG-19 model, the Grad-CAM model showed a maximum sensitivity value of 0.25, indicating more consistent reflections of predictions with less variation in explanations. On the other hand, the maximum sensitivity value of the Gradient Inputs model for VGG-19 was 0.30; this suggests higher variance in explanations around data points, potentially leading to less reliable insights at certain points. In conclusion, our findings show that lower maximum sensitivity values for both Grad-CAM and Gradient Inputs explainability models contribute to more successful and reliable explanations. A reduced precision value means greater consistency in descriptions around data points, increasing the reliability of guidance and facilitating a clearer understanding of model decisions.

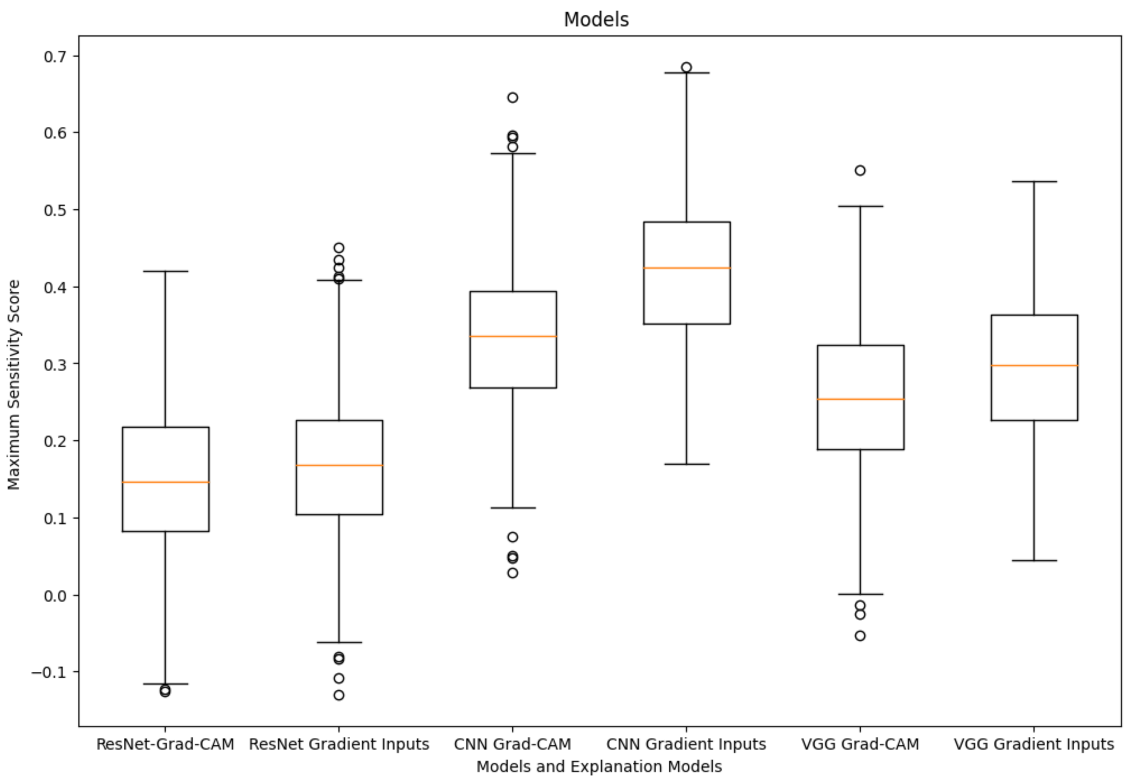

Figure 23. Maximum Sensitivity scores for deep learning models to detect estrus.

4.2.3. Complexity

The interpretability of deep learning models is critical to understanding the model's internal mechanisms and trusting its decisions. Therefore, we evaluated the explainability models of Grad-CAM and Gradient Inputs on a CNN model and a ResNet model and analyzed their complexity. As seen in Figure 24; the complexity value of the Grad-CAM explainability model applied to the CNN model was calculated as 9.76. This value shows that the Grad-CAM model structures the explanations of the CNN model's predictions in a rather complex way. In this case, the explanations may become difficult to understand, preventing the user from fully understanding the model's decisions. For the same CNN model, the complexity value of the Gradient Inputs explainability model was calculated as 9.35. This result shows that the Gradient Inputs model structures the explanations for the CNN model's predictions in a slightly less complex way. A lower complexity value indicates that the explanations may be more understandable and useful for better understanding the model's decisions. The complexity value of the Grad-CAM explainability model applied on the ResNet model was determined as 10.80. This value indicates that the Grad-CAM model structures the explanations for the predictions of the ResNet model in a very complex way. The complexity value of the Gradient Inputs explainability model for the ResNet model was calculated as 8.16. This result indicates that the Gradient Inputs model structures the explanations for the predictions of the ResNet model in a less complex way. The complexity value of the Grad-CAM explainability model applied on the VGG-19 model was determined as 9.80. This value indicates that the Grad-CAM model structures the explanations for the predictions of the VGG-19 model in a very complex way. The complexity value of the Gradient Inputs explainability model for the VGG-19 model was calculated as 8.75. A lower complexity value indicates that the explanations are more understandable and accessible to users. While Gradient Inputs provide a basic understanding of importance, Grad-CAM goes one step further to localize importance in intermediate feature maps. Therefore, the Grad-CAM explainability model has a more complex structure than the Gradient Inputs model.

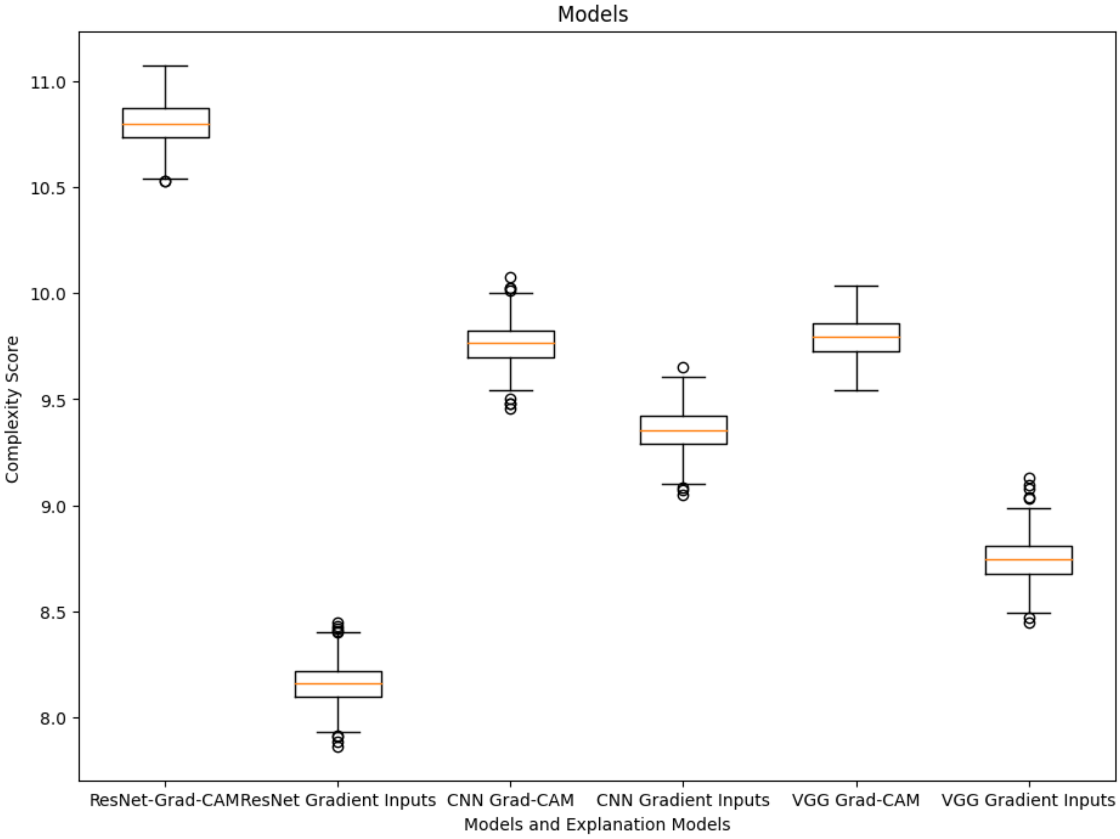

Figure 24. Complexity scores for deep learning models to detect estrus.

4.3. Result of Explainability Models

Grad-CAM is specifically designed for CNN. This approach generates a crucial feature map by utilizing gradients derived from class scores. Essentially, it employs weighted gradients to identify the impact of class scores on specific feature maps and quantifies this influence. Typically, these gradients are computed based on the outputs of the last convolutional layer, and a weighted feature map is constructed accordingly. Grad-CAM excels in highlighting the focused area on the map. In contrast, the "Gradient*Input" method involves multiplying gradients with input data to determine the model's classification decision. The fundamental concept is to element-wise multiply gradients with input data to discern the model's decision for a particular class. This method is specifically employed to identify the features that the model emphasizes during classification. The multiplication of class score gradients with input data results in what is often referred to as a class activation map. Grad-CAM proves effective in elucidating the model's predictions, providing a more reliable explanation of its decisions. Notably, Grad-CAM attains lower maximum sensitivity values, particularly for the ResNet model, signifying more consistent and precise explanations. However, Grad-CAM structures annotations in a complex manner, which may occasionally pose challenges in understanding them.

Table 2. Success rates for Explainability models.

Model	**XAI METHOD**	**FAITHFULNESS SCORE**	**MAXIMUM SENSITIVITY SCORE**	**COMPLEXITY SCORE**
CNN	Grad-CAM	0.33	0.32	9.76
CNN	Gradient Inputs	0.26	0.42	9.35
RESNET	Grad-CAM	0.2	0.15	10.8
RESNET	Gradient Inputs	0.15	0.17	8.16
VGG-19	Grad-CAM	0.28	0.25	9.8
VGG-19	Gradient Inputs	0.13	0.30	8.75

On the other hand, Gradient Inputs yield higher maximum sensitivity values, especially for the CNN model, indicating less precise and consistent annotations. Additionally, Gradient Inputs offer less intricate explanations, enhancing their overall comprehensibility. Nevertheless, Gradient Inputs present lower faithfulness values compared to Grad-CAM, suggesting weaker explanations of the model's predictions. The complexity values of Grad-CAM and Gradient Inputs play a role in the interpretability of explanations, with lower complexity values facilitating easier understanding of the explanations.

CHAPTER 5

DISCUSSION

This study aimed to determine the most suitable time for artificial insemination by determining the start times of the reproductive cycles of cows. Accurate determination of estrus periods aims to help farm owners avoid economic losses. Traditional estrus detection methods usually involve observing physical movements. In this study, estrus detection was aimed with artificial intelligence. However, this study aims to obtain more precise results by using XAI methods, unlike the methods commonly used in the literature. This can help farm owners increase productivity and use their resources more effectively, resulting in better results for less cost.

Explaining which features the developed models focus on when deciding estrus from mounting movement is one of the main differences of the study. This approach aims to provide farm owners with a more transparent and understandable estrus detection. When previous studies in the literature are examined as seen in Table 3, it is seen that Memedova and Keskin achieved 98% success with the fuzzy logic model they developed (Memedova et al. 2011). In Yıldız's doctoral dissertation, 97% success was achieved with the artificial neural network model that used seasonal data in addition to physical movements (Yıldız et al. 2022). The 94% success achieved by Arago and his team with models trained to detect estrus in cows is another important example of success in this field (Arago et al. 2020).

Table 3. Comprasion of studies for estrus detection.

Study	**Hardware**	**Software**	**Accuracy(%)**	**Explainability**
(Memmedova et al. 2011)	Pedometer	Fuzzy Logic Model	%98	NA
(Yıldız et al. 2022)	Pedometer	Artificial Neural Network	%97	NA
(Actimoo. 2023)	Pedometer	*	%80	NA
(Estrotect. 2023)	None	None	*	NA
(Arago et al. 2020).	Camera	Artificial Neural Network	%94	NA
Proposed CNN Model	Camera	Artificial Neural Network	%98	Grad-Cam, Gradient Inputs
Proposed ResNET Model	Camera	Artificial Neural Network	%99.18	Grad-Cam, Gradient Inputs
Proposed VGG-19 Model	Camera	Artificial Neural Network	%99	Grad-Cam, Gradient Inputs
Proposed YOLO-v5 Model	Camera	Artificial Neural Network	%98	NA

Table 4. Explainability studies in farming.

Study	Product	Model	Explainability	Explainability Metric
(Quach et al. 2024)	Tomato	DenseNet201, NasNetMobile, Xception, InceptionV3, MobileNet, EfficientNetV2	Grad-CAM	NA
(Siwar et al. 2023)	Potato	Faster RCNN	D-RISE	NA
(R. S et al. 2022)	Crop	Inception V3, ResNet	LIME	NA
Proposed CNN Model	Cow	CNN	Grad-CAM, Gradient Inputs	Faithfulness, Maximum Sensitivity, Complexity
Proposed ResNET Model	Cow	ResNET	Grad-CAM, Gradient Inputs	Faithfulness, Maximum Sensitivity, Complexity
Proposed VGG-19 Model	Cow	VGG-19	Grad-CAM, Gradient Inputs	Faithfulness, Maximum Sensitivity, Complexity

There have not yet been sufficient studies on explainability in the field of livestock farming in artificial intelligence studies. However, when studies on XAI in agriculture are examined, it is seen that technological progress has significant potential as seen in Table 4.

In the study conducted by Quach (Quach et al. 2024) and his team, they trained deep learning models by classifying the tomato damage dataset into categories such as 'Immature', 'Ripe', 'Old' and 'Damaged'. Grad-CAM XAI algorithm was used to explain the results of the deep learning model. This method has been used by identifying image features to evaluate the image recognition ability of deep learning or black box models. The research aims to evaluate the reliability of the model. While each trained model learned the characteristics of the damaged area, some models also recognized the background and healthy part of the tomato. However, some models have shown unreliable characteristics for the old label. Some other models could not detect image features in the damaged fruit area.

In the study of Siwar et al., an XAI method focusing on potato disease detection was proposed (Siwar et al. 2023). This method uses a perturbation-based approach to clarify both localization and classification aspects of potato leaf diseases.

In the study of R. S et al. the effects of various diseases of plants and climatic conditions were examined (R. S et al. 2022). This study performed damage detection using deep learning techniques and tried to understand the decision-making processes of deep learning models with explainability models such as Grad-CAM and LIME. However, in our study, estrus detection on cows was performed with deep learning models. As in other studies, different explainability models were trained on deep learning models, revealing the black box behind the models. The biggest difference that distinguishes our study from other studies is the evolution of explainability models. Faithfulness, Maximum Sensitivity and Complexity metrics were compared with how well the explainability models explained.

CHAPTER 6

CONCLUSION

This thesis presents an innovative approach utilizing machine learning and Explainable Artificial Intelligence (XAI) for detecting estrus behavior in cows. While previous studies have successfully identified estrus behaviors using machine learning, the black box of machine learning remains unexplained. Addressing this gap is deemed valuable, and the development of an algorithm capable of providing such insights holds significant importance. In livestock production, various methods exist for estrus detection, including wearable devices resembling pedometers or estrus patches. However, these commercial wearables have limitations such as the need for one device per animal, cost considerations, dependence on environmental factors, and limited lifespan. This study proposes a deep learning-based system aimed at mitigating the drawbacks of current methods and contributing to automated animal management.

In the proposed approach, deep learning-based detection of the mounting movement, which is the beginning of the estrus period, is performed and then helps us understand which features deep learning models make decisions with XAI methods. Then, the success of XAI methods is compared using metrics. In the proposed approach, 99% accuracy has been achieved with the ResNet transfer learning technique, and thanks to the Grad-CAM XAI model, it is observed that the detection of the mounting movement exhibited during the estrus period is made from the position characteristics of the cows' udder and back area.

This research provides a solution to automate livestock farming management that addresses the challenging and time-consuming limitations of traditional wearable systems, highlighting the critical importance of accurately determining the reproductive cycles of animals through artificial intelligence applications. This approach provides an image-based estrus period detection method that contributes to the literature and is designed to support automation and productivity increase in the livestock industry. In conclusion, this research proposes a sharp solution to improve livestock farming management, providing a method that can effectively detect estrous periods. It is poised to promote automation and productivity growth in the livestock sector. Image-based approach, XAI in animal husbandry, and evaluation metrics of XAI contribute to the literature.

REFERENCE

ActimooTM Actimoo Estrus Detection System. Available online: https://www.actimoo.com (accessed on 1 January 2023).

Andersson, L. & Okada, Hironao & Miura, Ryotaro & Zhang, Yi & Yoshioka, Koji & Aso, Hiroshi & Itoh, Toshihiro. (2016). Wearable wireless estrus detection sensor for cows. Computers and Electronics in Agriculture. 127. 101-108. 10.1016/j.compag.2016.06.007.

Arago, N.M.; Alvarez, C.I.; Mabale, A.G.; Legista, C.G.; Repiso, N.E.; Robles, R.R.A.; Amado, T.M.; Romeo, L.J., Jr.; Thio-ac, A.C.;Velasco, J.S. Automated Estrus Detection for Dairy Cattle through Neural Networks and Bounding Box Corner Analysis. Int. J.Adv. Comput. Sci. Appl. 2020,11, 935.

Araújo, S.O.; Peres, R.S.; Barata, J.; Lidon, F.; Ramalho, J.C. Characterising the Agriculture 4.0 Landscape—Emerging Trends,Challenges and Opportunities. Agronomy 2021,11, 667.

Arıkan, İbrahim, Tolga Ayav, Ahmet Çağdaş Seçkin, and Fatih Soygazi. 2023. "Estrus Detection and Dairy Cow Identification with Cascade Deep Learning for Augmented Reality-Ready Livestock Farming" Sensors 23, no. 24:9795. https://doi.org/10.3390/s23249795

Bhatt, Umang & Weller, Adrian & Moura, Jose. (2020). Evaluating and Aggregating Feature-based Model Explanations. 2988-2994. Calving and estrus detection in dairy cattle using a combination of indoor localization and accelerometer sensors, Computers and Electronics in Agriculture, Volume 168, 2020,

Chung, Y. & Choi, D. & Choi, Hyeongmo & Park, Daisy & Chang, H.-H & Kim, S.. (2015). Automated Detection of Cattle Mounting using Side-View Camera. KSII Transactions on Internet and Information Systems. 9. 3160-3177. 10.3837/tiis.2015.08.024.

Dallago, G.M.; Wade, K.M.; Cue, R.I.; McClure, J.T.; Lacroix, R.; Pellerin, D.; Vasseur, E. Keeping Dairy Cows for Longer: ACritical Literature Review on Dairy Cow Longevity in High Milk-Producing Countries. Animals 2021,11, 808.

Estrotect TM|Breed with Confidence. Available online: https://estrotect.com/ (accessed on 1 January 2023).

Fricke, P.M.; Carvalho, P.D.; Giordano, J.O.; Valenza, A.; Lopes, G.; Amundson, M.C. Expression and Detection of Estrus in Dairy Cows: The Role of New Technologies. Animal 2014,8, 134–143.

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. The Journal of physiology, 160(1), 106.

Jun Wang, Yifei Si, Jianping Wang, Xiaoxia Li, Kaixuan Zhao, Bo Liu, Yu Zhou, Discrimination strategy using machine learning technique for oestrus detection in dairy cows by a dual-channel-based acoustic tag, Computers and Electronics in Agriculture, Volume 210, 2023, 107949, ISSN 0168-1699, https://doi.org/10.1016/j.compag.2023.107949.

K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

LeCun, Y., Bengio, Y. and Hinton, G. (2015) Deep Learning. Nature, 521, 436- 444. http://dx.doi.org/10.1038/nature14539

Li, Z.; Liu, F.; Yang, W.; Peng, S.; Zhou, J. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects.IEEE Trans. Neural Netw. Learn. Syst. 2021,33, 6999–7019.

Liu, Y.; Ma, X.; Shu, L.; Hancke, G.P.; Abu-Mahfouz, A.M. From Industry 4.0 to Agriculture 4.0: Current Status, EnablingTechnologies, and Research Challenges. IEEE Trans. Ind. Inform. 2021,17, 4322–4334.

Memmedova, N.; Keskin, İ. Oestrus Detection by Fuzzy Logic Model Using Trait Activity in Cows. Kafkas Üniversitesi Vet.Fakültesi Derg. 2011,17, 1003–1008.

Quach, Luyl Da & Nguyen, Khang & Nguyen Quynh, Anh & Ngoc, Hoang & Nghe, Nguyen. (2024). Tomato Health Monitoring System: Tomato Classification, Detection, and Counting System Based on YOLOv8 Model with Explainable MobileNet Models using Grad-CAM++. IEEE Access. PP. 1-1. 10.1109/ACCESS.2024.3351805.

R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh and D. Batra, "Grad-

CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 618-626, doi: 10.1109/ICCV.2017.74.

R. S and I. M, "Explainable AI for Crop disease detection," 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2022, pp. 1601-1608, doi: 10.1109/ICAC3N56670.2022.10074303.

Redmon, Joseph & Divvala, Santosh & Girshick, Ross & Farhadi, Ali. (2016). You Only Look Once: Unified, Real-Time Object Detection. 779-788. 10.1109/CVPR.2016.91

Remnant, J.G.; Green, M.J.; Huxley, J.N.; Hudson, C.D. Associations between Dairy Cow Inter- Service Interval and Probability ofConception. Theriogenology 2018,114, 324–329.

Roelofs, J.; López-Gatius, F.; Hunter, R.H.F.; van Eerdenburg, F.J.C.M.; Hanzen, C. When Is a Cow in Estrus Clinical and Practical Aspects. Theriogenology 2010, 74, 327–344.

S. Benaissa, F.A.M. Tuyttens, D. Plets, J. Trogh, L. Martens, L. Vandaele, W. Joseph, B. Sonck, 105153, ISSN 0168-1699, https://doi.org/10.1016/j.compag.2019.105153.

Simonyan, Karen & Zisserman, Andrew. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv 1409.1556.

Siwar Bengamra & Zagrouba, Ezzeddine & Bigand, André. (2023). Explainable AI for Deep Learning Based Potato Leaf Disease Detection. 1-6. 10.1109/FUZZ52849.2023.10309803.

Springenberg, Jost & Dosovitskiy, Alexey & Brox, Thomas & Riedmiller, Martin. (2014). Striving for Simplicity: The All Convolutional Net.

Webster, J. Understanding the Dairy Cow; John Wiley & Sons: Hoboken, NJ, USA, 2020; ISBN 978-1-119-55022-8.

Yeh, Chih-Kuan, Cheng-Yu Hsieh, Arun Sai Suggala, David I. Inouye and Pradeep Ravikumar. “On the (In)fidelity and Sensitivity of Explanations.” Neural Information Processing Systems (2019).

Yıldız, A.K.; Özgüven, M.M. Determination of Estrus in Cattle with Artificial Neural Networks Using Mobility and Environmental Data. J. Agric. Fac. Gaziosmanpaşa Univ. (JAFAG) 2022, 39, 40–45.