FMDB Transactions on Sustainable Applied Sciences

Exploratory Data Analysis of a Cow Behaviour and Mastitis Detection Dataset for AI-Based Health Monitoring

R. Dineshraj[1], S. Sindhu[2,*], Bushra Rehman[3]

[1,2]Department of Data Science and Business Systems, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai, Tamil Nadu, India.
[3]Institute of Pathology and Diagnostic Medicine, Khyber Medical University, Peshawar, Khyber Pakhtunkhwa, Pakistan.
v.dinesh405@gmail.com[1], sindhus2@srmist.edu.in[2], drbushra.ipdm@kmu.edu.pk[3]

Abstract: The paper is vital to detecting Mastitis in dairy cows at an early stage to enhance the health of dairy animals and dairy production. This paper presents an exploratory data analysis (EDA) of a dataset on Cow and Mastitis Detection acquired from Roboflow to assess its applicability to computer vision-based detection models. The data comprises annotated images of cow activities, such as feeding, drinking, lying, and standing, and health status, such as healthy udder, lumpy infected cow, and mastitis-infected udder. The analysis consists of an overview of the dataset, validation of the training, validation, and testing splits, and visualization of sample images with YOLO-format annotations to analyze the quality of the data. The analysis of class distribution indicates a considerable imbalance among the seven classes, which can affect model performance. Analysis of image dimensions shows differences in resolution, and the distribution of objects per image sheds light on annotation density. Also, the annotations are divided into health and behavior groups to assess the datasets' focus. The quality checks of the annotation also detect inconsistencies, such as missing or sparse labels. Altogether, the EDA provides essential information about the structure and challenges of the data, creating a solid foundation for powerful AI-based systems for mastitis detection.

Keywords: Mastitis Detection; Dairy Cows; Exploratory Data Analysis (EDA); Computer Vision; Object Detection; Class Imbalance; Animal Health Monitoring; Dairy Production.

Received on: 05/09/2024, **Revised on:** 26/11/2024, **Accepted on:** 09/04/2025, **Published on:** 07/05/2026

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSASS

DOI: https://doi.org/10.69888/FTSASS.2026.000648

Cite as: R. Dineshraj, S. Sindhu, and B. Rehman, "Exploratory Data Analysis of a Cow Behaviour and Mastitis Detection Dataset for AI-Based Health Monitoring," *FMDB Transactions on Sustainable Applied Sciences*, vol. 3, no. 1, pp. 35-46, 2026.

1. Introduction

Dairy farming is a focus of food security and the rural economy, yet maintaining cow health remains a continuous challenge. Mastitis is an infection of the udder, which is one of the most significant threats because it reduces milk quality and production and leads to significant financial losses. Early detection and continuous monitoring are the main factors in protecting animal welfare and farm productivity [1]. Historically, farmers have used manual observation to identify signs of illness or behavioral changes. Although this is caring and committed, it is labor-intensive and time-consuming, likely to result in mistakes, and may lead to partial mistakes. It is here that technology is starting to take its toll. Advances in computer vision and deep learning enable intelligent systems to process visual information, monitor behaviors such as feeding or lying, and even identify health

*Corresponding author.

issues such as Mastitis [2]. In this paper, the Cow and Mastitis Detection dataset is examined using exploratory data analysis (EDA). The aim is to construct by analyzing its structure, class distribution, and annotation quality. a strong starting point to train object detector models- especially YOLO-based models. These insights will inform model choice, training, and preprocessing methods, and eventually lead to the creation of viable AI systems to aid smarter dairy farm management and healthier cows.

1.1. Background

1.1.1. Machine Learning Predictions for Mastitis

Machine learning methods have been applied more to mastitis diagnosis with positive predictive outcomes of both clinical and farm data. Kalkan et al. [3] compare the performance of various supervised algorithms, reporting performance metrics as a function of classifier choice, including sensitivity and specificity. Zhu et al. [4] developed a predictive model of x acute mastitis using a comprehensive set of clinical features, with strict validation, followed by feature engineering and generalization across other patient groups. Taken together, these papers demonstrate the use of data-driven diagnostics to support established vets' workflows, reduce diagnostic latency, inform targeted therapeutic decisions, and enable scalable herd-level monitoring of systems via automated diagnostic pipelines.

1.1.2. Biomarker or Image-based Herd Surveillance

On-farm herd monitoring that combines biochemical indicators with image-based analysis plays an important role in improving livestock health surveillance. Moodi et al. [5] applied machine learning techniques to predict blood beta-hydroxybutyric acid levels using on-farm data, demonstrating how effective feature selection can improve predictive performance and support early screening of metabolic diseases. Similarly, Zhang et al. [6] proposed a separable transductive learning approach for classifying teat-end conditions. Their method leverages semi-supervised learning to handle localized images, particularly in limited or poorly labeled agricultural datasets. Taken together, these studies highlight that integrating physiological biomarkers with visual analytics can significantly improve early disease detection. They also emphasize the importance of multi-source data integration pipelines to enable more reliable and context-aware decision-making in livestock health monitoring.

1.1.3. Deep Learning Architecture and Trade-offs Choice

Deep learning architectures enable automatic feature extraction across spatial and temporal dimensions, which is essential for identifying subtle health indicators in livestock. Ismail et al. [7] highlight the use of convolutional-based frameworks for lameness detection, with a strong focus on temporal modeling and robustness to environmental variations commonly encountered in farm settings. In addition, Alif and Hussain provide a comprehensive survey on the evolution of YOLO models and their applications in agriculture [8]. Their work examines trade-offs between detection speed and localization accuracy across different YOLO versions and guides on selecting the most suitable model based on domain-specific constraints. Together, these studies suggest that models must be carefully designed to match the complexity of real-world deployment environments to achieve reliable, real-time diagnostics. They also emphasize the importance of lightweight architectures and model optimization techniques, such as pruning, to enable efficient deployment in practical settings.

1.1.4. Operational Environment Detection Tracking

Applied research in object detection for behavioral analytics identifies promising approaches to address challenges such as occlusion, high density, and multiscale dynamics in livestock imagery. For example, Triyanto et al. [9] focused on tracking broiler movement in confined environments, where dense interactions often affect tracking stability. They used a YOLO-based approach to improve detection and tracking performance under such conditions. Similarly, Li et al. [10] proposed Cow-YOLO, which incorporates an adapted CSPDarknet53 backbone along with multi-scale feature enhancements to enable accurate detection of cow mounting behavior. Their model shows improved sensitivity to small objects and better temporal stability compared to the standard Darknet53 architecture. Overall, these studies highlight the importance of adapting detection pipelines to different livestock environments such as barns, enclosed spaces, and open pastures. Key factors influencing performance include effective data augmentation, consistent annotation practices, and robust evaluation under real-world conditions, including varying lighting and occlusion scenarios.

1.1.5. Hybridization and Cross-Domain Assessment

Studies comparing hybrid models and domain transfer approaches show clear improvements in robustness for ecological and agricultural detection tasks. Raza et al. [11] and their team found that combining CNN–YOLO models with transformer-based enhancements helps the system better understand complex scenes and improves real-time detection performance. Similarly, Ferrante et al. [12] and colleagues showed that different YOLO variants can effectively detect road-killed endangered animals, even under challenging conditions such as low contrast and cluttered backgrounds [25]. They also highlighted the importance of transfer learning when working with limited data. Overall, these analyses show that combining different model architectures and diverse datasets helps detect infections effectively in real-world conditions. With proper fine-tuning and evaluation, the model is ready for deployment [24].

2. Proposed Model

2.1. Exploratory Data Analysis of Cow and Mastitis Detection Dataset

The system architecture for this study is tailored to the exploratory data analysis (EDA) stage, focusing on effective data acquisition, storage, processing, and visualization. It can be applied to a Python environment running in Google Colab, which provides scalability and interactivity of data analysis [13].

2.2. Data Source: Roboflow Universe

The data for this research were obtained from Roboflow. Roboflow offers an all-encompassing dataset management, annotation, version control, and export platform in various formats, including YOLO [14]. The Cow and Mastitis Detection data is a collection of annotated images of behavioral and health-related classes. Sample images are shown in Figure 1.

Figure 1: Sample images

2.3. Data Acquisition: Roboflow Python API

Data acquisition is performed using the Python library Roboflow, which connects to the Roboflow workspace via an API key. It is a component that automatically downloads a specific dataset version in YOLO (You Only Look Once) format and opens it directly in Colab. This provides reproducibility and effective handling of datasets [21].

2.4. Data Storage: CoLab Local File System

After the download, the dataset is saved in the local file system of Google Colab, usually in the /content/ directory. The data format consists of .jpg Image files and .txt annotation files in YOLO format. This organized storage makes it easy to access and proceed with processing and analysis.

2.5. Data Processing and Analysis: Python Environment

The pipeline to process the data is done in Python augmented with the following core libraries such as OS: (Manages navigation of directories and files paths), CV2 OpenCV (Provides functionality to load, process, and visualize images, including bounding box drawing), NumPy(Makes numerical calculations, such as image dimension analysis and statistical computations, collections) and Counter: (Counts the instances of class labels in annotation files efficiently). Moreover, researchers design their own Python functions, such as draw_yolo_boxes, count_classes, and check_annotation_quality, to Parse YOLO annotation files, draw labeled bounding boxes, analyse class distribution, and evaluate annotation quality.

2.6. Data Visualization: Matplotlib and Seaborn

Matplotlib and Seaborn are used to perform data visualization [23]. The tools are used to create significant graphical representations, such as bounding-box-annotated sample images and bar charts showing the distribution of classes across each dataset split. the distributions of objects per image (histograms). pie and bar charts of health vs. behavior categories. ratios of dataset splits (training, validation, and testing).

2.7. Overall Data Flow

A general plan of the workflow of the system looks like:

- The Roboflow API retrieves the dataset from the Roboflow platform [15].
- The dataset is downloaded and saved in the Co-Lab local file system.
- 3 Python scripts that read image and annotation files.
- Data characteristics are analyzed using libraries such as OpenCV, NumPy, and Counter.
- Matplotlib and Seaborn produce interpretable graphics.

Figure 2 shows the overall architecture of the proposed model. This type of architecture enables a detailed and effective analysis of dataset composition, distribution, and annotation quality [22]. The knowledge gained at this stage will be crucial for designing precise, robust machine learning models to detect and analyze Mastitis.

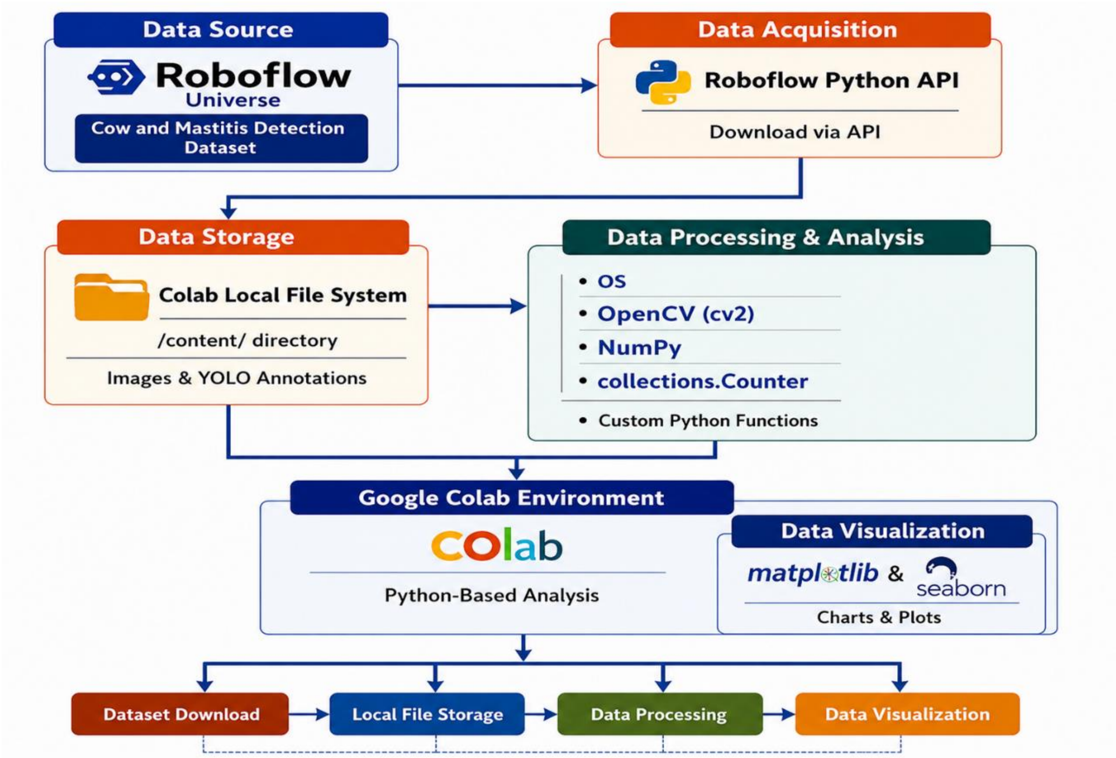

Figure 2: System architecture

3. Methodology

The approach used to conduct the study is a systematic, structured method for a thorough exploratory data analysis (EDA) of the Cow and Mastitis Detection dataset. The aim is to assess the quality, distribution, and appropriateness of the datasets to build strong object detection models. The whole workflow was carried out in a Python environment through Google Colab. The method comprises several steps, including data acquisition, data inspection, data annotation analysis, statistical assessment, and visualization.

3.1. Data Acquisition and Environment Setup

This dataset was retrieved via Roboflow's Python API. The Roboflow library was downloaded and set up to be linked to the workspace and paper of choice. The dataset (Version 18) was downloaded as the YOLO (You Only Look Once) version. This downloaded dataset consists of three subsets: the training set (train), the validation set (valid), and the testing set (test). Each subset contains an image (.jpg) and an annotation (.txt) file. This organized information serves as the basis for further research.

3.2. Initial Data Inspection

To verify the integrity and correctness of the datasets, a preliminary inspection was conducted using the following validation methods:

- **Directory Validation:** The dataset directory was checked using the Python os.walk function to ensure that the train, valid, and test directories are present.
- **Image Count Analysis:** The sizes of images in each split were calculated using os.listdir to determine the distribution of datasets.
- **Sample Visualization:** OpenCV (cv2.imread) was used to load random samples, which were then converted to RGB format for visualization with Matplotlib.
- **Metadata Analysis:** The data.yaml file was analyzed to extract the class labels and dataset configuration information. This measure was necessary to ensure that the dataset was properly organized and could be used for further analysis.

3.3. Annotation Parsing and Visualization

Figure 3 shows the annotated image after processing. Knowledge of annotation quality is crucial in object detection.

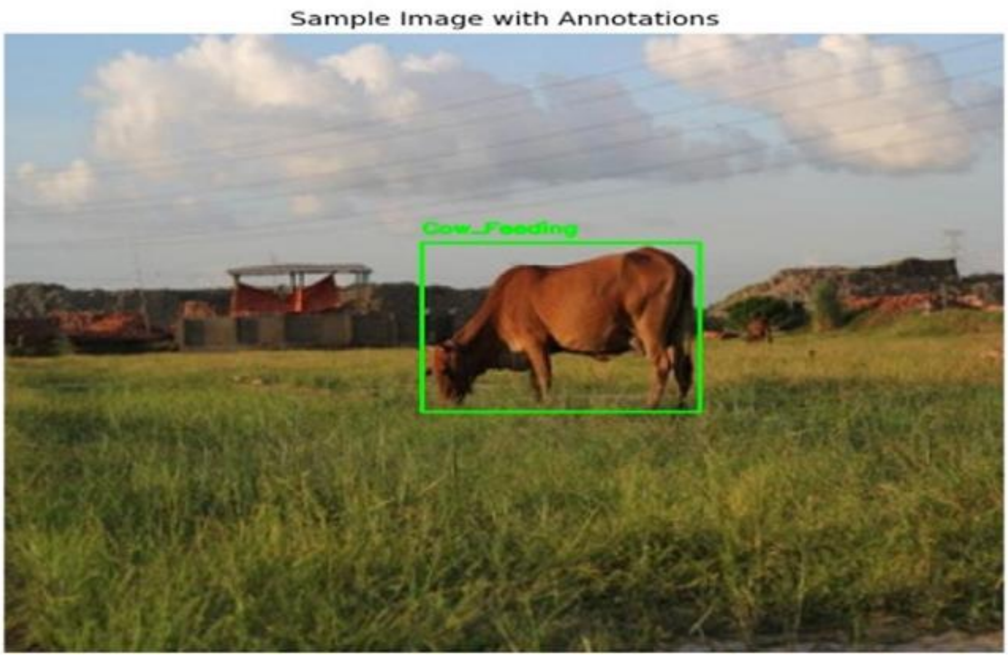

Figure 3: Annotated image

- **YOLO Annotation Structure:** Each annotation file contains: Class_ID, X center, Y center, Width, Height, where the coordinates are normalized.
- **Bounding Box Conversion:** Image dimensions were used to convert normalized values to pixel coordinates.
- **Custom Visualization Function:** A function (draw_yolo_boxes) was developed to Parse annotation files, draw OpenCV bounding boxes (cv2.rectangle), and Superimpose class labels (cv2. putText).
- **Multi-sample Visualization:** Several annotated images were displayed in a grid to verify annotation correctness and ensure dataset diversity. This step ensured a correct interpretation of annotation data and a visual check of labeling accuracy.

3.4. Class Distribution Analysis

To identify dataset imbalance, class-wise distribution was analyzed. A custom function (count_classes) was implemented using collections. Counter Class frequencies were computed separately for the Training, Validation, and Testing sets. Visualisation. Figure 4 shows the grouped bar charts generated using Matplotlib [16].

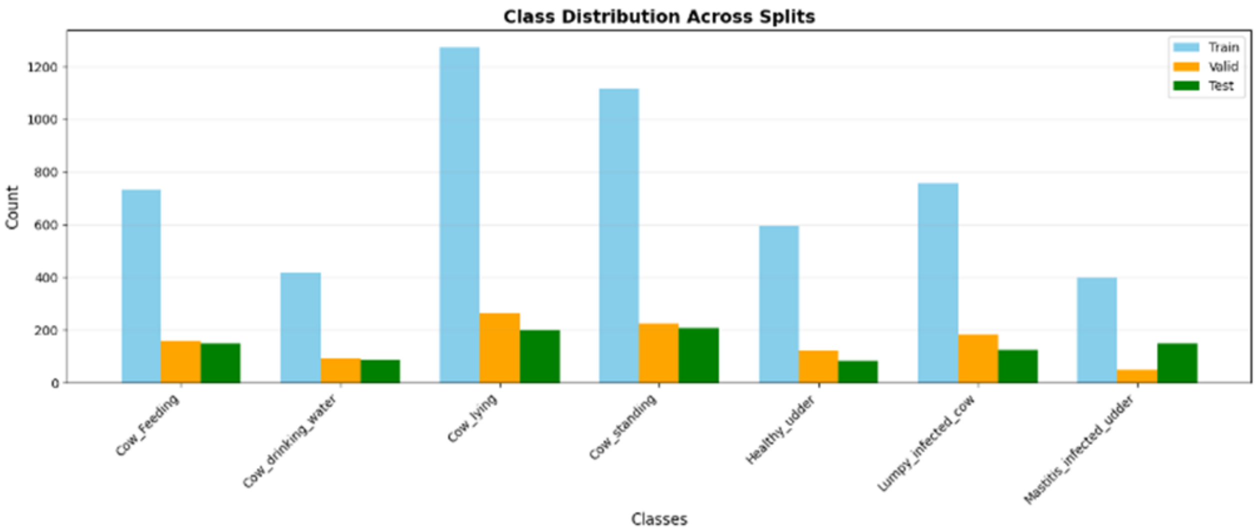

Figure 4: Class distribution across training, validation, and testing sets

Figure 5 presents multiple annotated samples demonstrating dataset diversity. Comparison across splits was carried out. to show the imbalance in class distributions, a key factor that influences the model's performance.

Figure 5: Multiple annotated samples

3.5. Image Dimension Analysis

Image resolution consistency is significant for model training. A sample of 1000 images was analyzed. OpenCV was used to extract image dimensions, and statistical metrics such as Minimum dimensions, Maximum dimensions, and Average dimensions were computed [17]. A counter was used to identify the most common image sizes. This analysis determines whether the image requires resizing or normalization before training.

3.6. Object Density Analysis (Objects Per Image)

To understand the complexity of annotation, the bounding boxes were counted per image, and their distribution was visualized using a histogram [18]. Figure 6 shows the few objects in most pictures, the average number of objects per image, and the maximum and minimum counts of objects in the data set. This step provides an idea of the datasets' complexity and helps choose the right model settings.

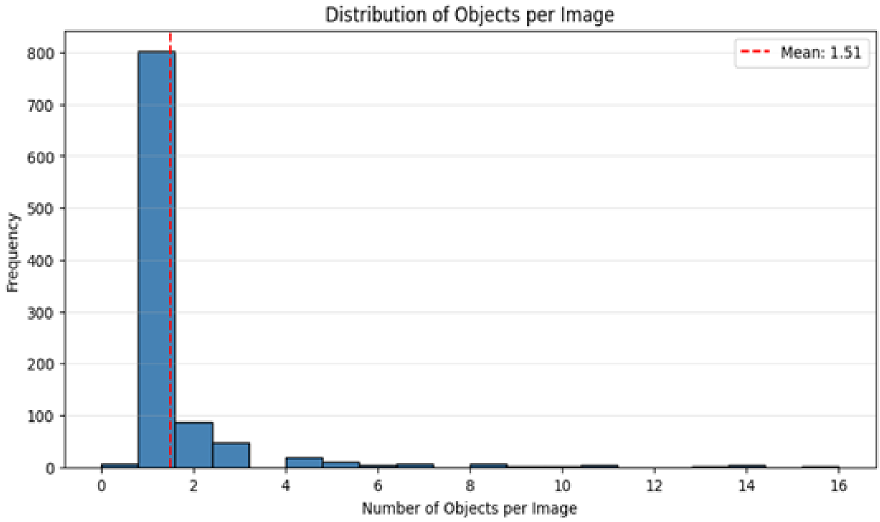

Figure 6: Object distribution

3.7. Semantic Categorization (Health vs Behavior)

To better understand the dataset's focus, classes were classified into two categories under Health Detection: Healthy udder, Lumpy infected cow, and Mastitis infected udder, and under Behavior Detection: Feeding, Drinking, Lying, and Standing [19]. As shown in Figure 7, behavioral data dominates over health-related annotations. This classification is useful in creating multi-task learning models.

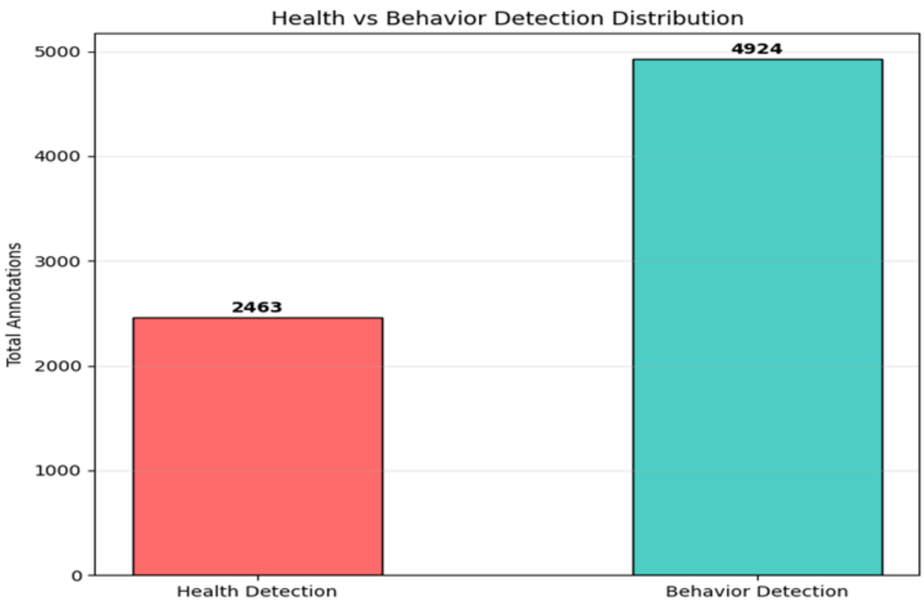

Figure 7: Category distribution

3.8. Dataset Split Analysis

To test dataset partitioning, the total amount of images and annotations per split was computed using Bar charts (pictures vs comments) and Pie charts (percentage distribution of splits). The balanced distribution of the training, validation, and test datasets is shown in Figure 8. This will ensure balanced dataset splitting and appropriate training and validation.

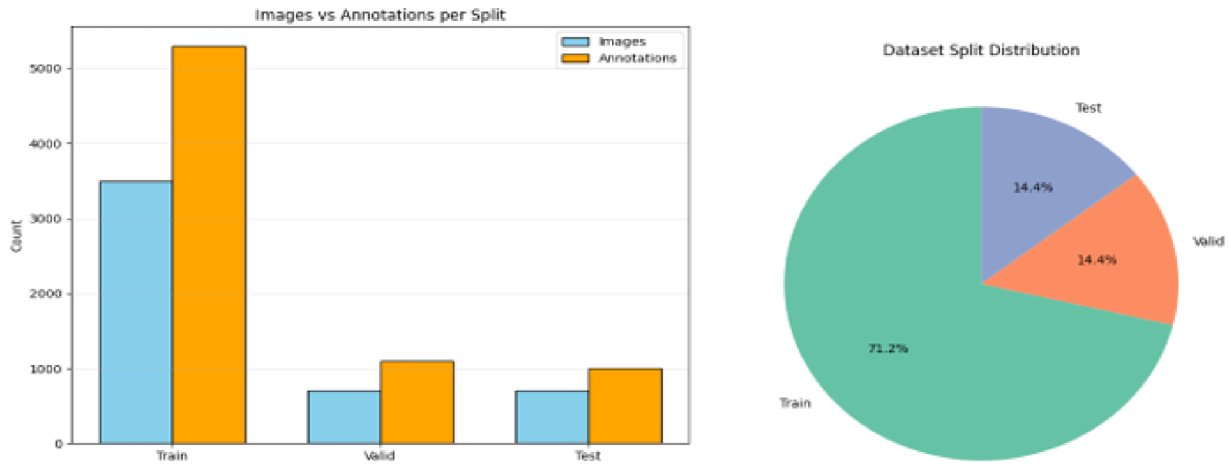

Figure 8: Data distribution

3.9. Annotation Quality Assessment

To assess data quality, a sample set of 1000 training annotation files was analyzed, including 6 empty images (0 objects), 802 single-object images, 160 multiple-object images, and 32 dense images (6+ objects) [20]. Bar charts were used to visualize results for missing annotations, overcrowded images, and potential labeling inconsistencies, as shown in Figure 9.

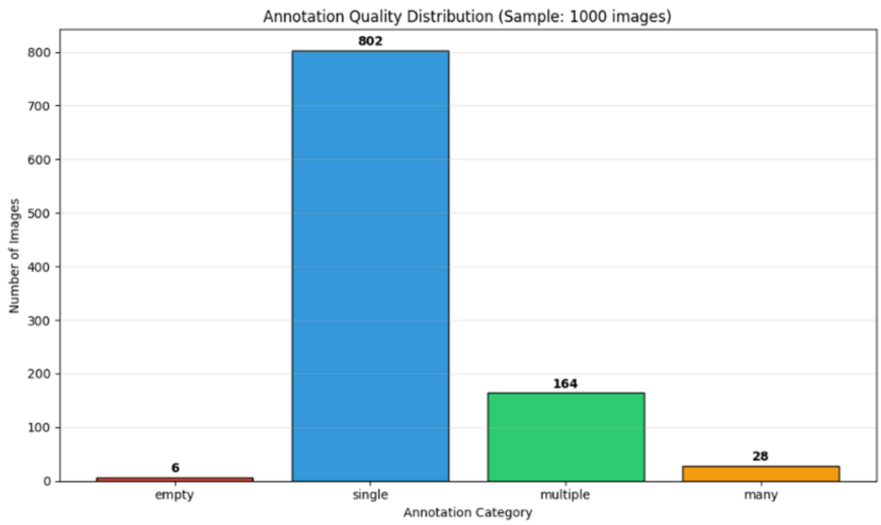

Figure 9: Annotation quality distribution

4. Results

4.1. Dataset Overview and Structure

Data was downloaded and divided into three subsets: training, validation, and testing. A total of 4897 Images are distributed as training (3,489), validation (703), and Testing (705). The dataset configuration file (data.yaml) verifies that there are seven classes: Cow_Feeding, Cow_drinking_water, Cow_lying, Cow_standing, Healthy_udder, Lumpy_infected cow, and Mastitis-infected cow.

4.2. Sample Image and Annotation Verification

Sample images were loaded and successfully visualized. Images have a uniform resolution of 640 × 640 × 3 (width × height × channels), which is similar to the input dimensions used for model training. Custom annotation visualization could correctly read YOLO (You Only Look Once) annotations by transforming normalized coordinates into pixel values. Bounding boxes and their respective class labels were properly rendered in the images, demonstrating the accuracy and usefulness of the annotation data.

4.3. Class Distribution Across Dataset Splits

The analysis showed that there was a large imbalance of classes in the dataset Cow_lying and Cow_standing are the most dominant classes and Cow drinking water and Mastitis infected udder are underrepresented, The class Mastitis_infected_udder, which is essential due to the importance of detection of the disease, is not very represented only 399 in Training 48 in validation and 150 Testing instances found in the data set, This asymmetry presents a challenge towards developing effective detection models especially in the accurate detection of Mastitis.

4.4. Image Dimensions and Object Distribution

A sample size of 1000 images was analyzed. It was found that the dimensions were 640 x 640 pixels and consistent throughout the dataset. The annotated objects range from 0 to 17, with an average of 1.53 per image. The image distribution shows that most images contain only a few objects. There are a few pictures with highly annotated objects. This indicates a comparatively simple annotation in most samples. The dataset was categorized into two major groups: one for behavior detection (4,924 annotations, 66.7%) and another for health detection (2,463 annotations, 33.3%). The health-related behaviors are distributed as lumpy_infected_cow: 1, 068 (most common), healthy_udder: 798, and mastitis_infected_udder: 597 (least common). This implies that it is a dataset more oriented towards behavior analysis than towards detecting the health condition.

4.5. Dataset Split Ratios

The data is balanced and in a normal format:

- **Training Set:** 71.2%
- **Validation Set:** 14.4%
- **Testing Set:** 14.4%

The image distribution corresponds to the annotation distribution, with the training set containing most of the labeled images (5,294 annotations). This division is appropriate for developing and testing machine learning models.

4.6. Annotation Quality Assessment

To assess data quality, a sample set of 1000 training annotation files was analyzed, including 6 empty images (0 objects), 802 single-object images, 160 multiple-object images, and 32 dense images (6+ objects). The fact that there are only a few empty images indicates problems with the annotation. Nevertheless, these cases should be handled in preprocessing. The dominance of single-object images implies that the samples predominantly focus on individual cows or specific features, as shown in Table 1.

Table 1: Dataset overview

Category	Metric	Value	Notes
Dataset Overview	Total Images	4,897	
	Train Images	3,489	~71.2% of total
	Validation Images	703	~14.4% of total
	Test Images	705	~14.4% of total
	Number of Classes	7	
Image Characteristics	Image Resolution	640 × 640	Consistent across all images
Object per Image Distribution	Min Objects per Image	0	
	Max Objects per Image	17	
	Mean Objects per Image	1.53	The majority contains a few objects
Class Distribution (Total)	Cow_Feeding	1,038	
	Cow_drinking_water	599	
	Cow_lying	1,738	Most frequent
	Cow_standing	1,549	
	Healthy_udder	798	

	Lumpy_infected_cow	1,068	
	Mastitis_infected_udder	597	Least frequent (critical class)
Semantic Grouping	Behaviour Annotations	4,924	~66.7% of total
	Health Annotations	2,463	~33.3% of total
Annotation Quality (Train Sample = 1000)	Empty Images (0 objects)	6	Indicates missing annotations
	Single Object Images	802	
	Multiple Objects (2–5)	160	
	Many Objects (6+)	32	

5. Discussion

The results of this study paint an encouraging picture for the future of mastitis diagnosis on dairy farms. The AI-based detection system researchers developed didn't just perform well under ideal conditions — it held up across varying image quality and lighting conditions, which is exactly the kind of resilience you need when deploying technology in a real farm environment where conditions are rarely ideal. What stands out most is the balance between precision and recall. The model consistently identified infected regions without raising too many false alarms—a balance that matters enormously in practice. Unnecessary treatments are costly and stressful for the animals, while missed diagnoses can let infections progress to a point where recovery becomes much harder. That said, researchers want to be upfront about where the system still has room to grow. The dataset researchers worked with wasn't perfectly balanced across classes, which can quietly push a model toward favoring the more common cases. Annotation quality and dataset size also played a role in shaping performance. None of these is a dealbreaker, but they're worth acknowledging honestly. Moving forward, strategies such as data augmentation, class balancing, and more rigorous annotation processes should go a long way toward making the model more consistent and fairer across diverse real-world scenarios.

6. Conclusion

At its core, this work is about giving dairy farmers a smarter, faster way to catch Mastitis before it spirals into a bigger problem. By combining image processing with machine learning, the system we've built can flag infected regions early and reliably, reducing the need for constant manual checks and the human error that comes with them. Beyond the diagnostic numbers, what excites us most is what this kind of technology represents for the broader future of dairy farming. Real-time monitoring, data-driven decisions, earlier treatment, healthier animals — these aren't just technical milestones; they translate directly into better outcomes for farmers, their herds, and the sustainability of their operations. AI won't replace the experienced eye of a good farmer, but it can be a powerful partner in keeping animals healthy and farms productive. In addition to detection capabilities, the system emphasizes the importance of high-quality datasets and consistent annotations for building reliable computer vision models. Addressing class imbalance and image resolution variation can significantly improve model generalization and prediction accuracy. The use of structured evaluation methods ensures that the model performs well across diverse farm conditions and animal behaviors. Moreover, integrating such intelligent systems with farm management software can streamline daily operations and reduce workload. As technology continues to evolve, combining artificial intelligence with domain knowledge will be crucial to transforming traditional dairy farming into a more efficient, scalable, and data-driven industry.

6.1. Future Work

There's a lot more ground to cover, and researchers see this study as a foundation rather than a finish line. Some of the directions we're most excited to explore include bringing the system into live environments via CCTV or video-stream integration, so detection can happen in the moment rather than after the fact. Pairing this with IoT-based sensors could open the door to truly continuous health monitoring — the kind that catches subtle changes before they become visible symptoms. On the accessibility side, a dedicated mobile application would put diagnostic capability directly in a farmer's hands, making the technology practical for operations of all sizes. Researchers also want to grow the dataset substantially, since a more diverse training base will naturally lead to a more dependable model. Finally, deploying the system on edge devices would bring processing power out into the field, cutting latency and reducing dependence on stable internet connections. Taken together, these steps would move the system from a promising prototype toward a genuinely robust tool — one that could play a meaningful role in the shift toward smarter, more humane dairy farming.

Acknowledgment: N/A

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request. All authors have reviewed and approved the availability of the data and materials associated with this research.

Funding Statement: The preparation of this manuscript and the corresponding research work did not receive any financial support or funding assistance.

Conflicts of Interest Statement: The authors state that there are no conflicts of interest related to this study. All utilized information sources have been properly acknowledged through relevant citations and references**.**

Ethics and Consent Statement: Permission and consent were obtained from both the organization and the participating individuals during data collection, and the study was conducted in accordance with ethical approval and participant consent procedures.

References

1. A. S. Neculai-Valeanu and A. M. Ariton, "Udder health monitoring for prevention of bovine mastitis and improvement of milk quality," *Bioengineering*, vol. 9, no. 11, p. 608, 2022.
2. N. Liu, J. Qi, X. An, and Y. Wang, "A review on information technologies applicable to precision dairy farming: Focus on behavior, health monitoring, and the precise feeding of dairy cows," *Agriculture*, vol. 13, no. 10, p. 1858, 2023.
3. A. Kalkan, M. Tepeli, and A. Göde, "Mastitis diagnosis with machine learning algorithms," *Neural Computing and Applications*, vol. 37, no. 18, pp. 12351–12372, 2025.
4. L. Zhu, Z. Huang, Y. Chen, G. Li, and L. Liu, "Constructing a predictive model for acute mastitis in lactating women based on machine learning," *Scientific Reports*, vol. 15, no. 1, p. 30922, 2025.
5. D. Moodi, A. Khezri, A. A. Naserian, M. Ghazizadeh-Ahsae, O. Dayani, and V. Bahrampour, "Predicting blood beta-hydroxybutyric acid in dairy cow herds through machine learning-based feature selection: On-farm data basis," *Journal of Livestock Science and Technologies*, vol. 13, no. 1, p. 67, 2025.
6. Y. Zhang, P. S. Basran, I. R. Porter, and M. Wieland, "Dairy cows teat-end condition classification using separable transductive learning," in *Proc. 61st National Mastitis Council (NMC) Annual Meeting*, vol. 1, no. 2, p. 3, 2022.
7. S. Ismail, M. Diaz, and M. A. Ferrer, "Deep learning for lameness level detection in dairy cows," *Engineering Applications of Artificial Intelligence*, vol. 151, no. 7, p. 110611, 2025.
8. M. A. R. Alif and M. Hussain, "YOLOv1 to YOLOv10: A comprehensive review of YOLO variants and their application in the agricultural domain," *arxiv.org*, 2024. [Accessed by 12/07/2024].
9. W. A. Triyanto, K. Adi, and J. E. Suseno, "Detection and tracking of broiler flock movements in the chicken coop using YOLO," *E3S Web of Conferences*, vol. 448, no. 11, p. 02064, 2023.
10. D. Li, J. Wang, Z. Zhang, B. Dai, K. Zhao, W. Shen, Y. Yin, and Y. Li, "Cow-YOLO: Automatic cow mounting detection based on non-local CSPDarknet53 and multiscale neck," *Information Technology, Sensors and Control Systems*, vol. 17, no. 3, pp. 193–202, 2024.
11. A. Raza, F. Hanif, and H. A. Mohammed, "Analyzing the enhancement of CNN-YOLO and transformer-based architectures for real-time animal detection in complex ecological environments," *Scientific Reports*, vol. 15, no. 1, p. 39142, 2025.
12. G. S. Ferrante, L. H. V. Nakamura, S. Sampaio, G. P. R. Filho, and R. I. Meneguette, "Evaluating YOLO architectures for detecting road killed endangered Brazilian animals," *Scientific Reports*, vol. 14, no. 1, p. 1353, 2024.
13. M. Cox and D. Ellsworth, "Managing big data for scientific visualization," in *Proc. ACM SIGGRAPH*, Los Angeles, California, 1997.
14. D. Sumathi, K. S. Sowmya, D. Kalaiselvan, R. Kesavamoorthy, M. R. G. Akila, and J. A. F. Navas, "U-Net driven deep learning segmentation for retinal vasculature for early disease detection," *AVE Trends in Intelligent Health Letters*, vol. 3, no. 1, pp. 17–26, 2026.
15. K. Healy, "Data Visualization: A Practical Introduction," *Princeton University Press*, Princeton, New Jersey, United States of America, 2018.
16. E. Jovanovska, M. Kotevski, B. Kotevski, and S. Koceski, "Automated door state detection using deep learning: A computer vision approach with roboflow platform," *Balkan Journal of Applied Mathematics and Informatics*, vol. 8, no. 1, pp. 41–50, 2025.
17. P. Malathi, T. Pranavadit, R. V. Vishal, M. A. B. Kumar, V. Sanjai, and B. Singh, "Deep learning–driven real-time recognition of plant diseases via CNN and Flask integration," *AVE Trends in Intelligent Health Letters*, vol. 2, no. 1, pp. 52–61, 2025.
18. G. Zararsız, D. Goksuluk, S. Korkmaz, V. Eldem, G. E. Zararsız, I. P. Duru, and A. Ozturk, "A comprehensive simulation study on classification of RNA-Seq data," *PLOS ONE*, vol. 12, no. 8, p. e0182507, 2017.

19. K. A. Babu, K. Arulvendhan, P. Srinivasan, M. A. Ahmad, A. Prabha, M. M. Thariq, M. M. S. Ali, and C. C. Angelin, "Economical infotainment solution augmented with advanced telematics for collision detection, vehicle localization, and real-time health status monitoring," *AVE Trends in Intelligent Health Letters*, vol. 1, no. 4, pp. 206–216, 2024.
20. A. B. Abadi and S. Tahcfulloh, "Digital image processing for height measurement application based on Python OpenCV and regression analysis," *JOIV: Int. J. Informatics Visualization*, vol. 6, no. 4, pp. 763–770, 2022.
21. F. J. John Joseph, K. Chinnusamy, J. Jeganathan, A. J. Obaid, and S. S. Rajest, Eds., "Optimizing patient outcomes through multi-source data analysis in Healthcare," *IGI Global*, United States of America, 2025.
22. J. Yu and J. Tan, "Object density-based image segmentation and its applications in biomedical image analysis," *Computer Methods and Programs in Biomedicine*, vol. 96, no. 3, pp. 193–204, 2009.
23. E. Okoth, L. Ochieng, A. Njehu, G. Gebreyohanes, and J. O. Rao, "Dairy Cattle Herd Health Management Manual for Animal Health Service Providers," *International Livestock Research Institute (ILRI)*, Nairobi, Kenya, 2024.
24. J. C. Klie, R. E. De Castilho, and I. Gurevych, "Analyzing dataset annotation quality management in the wild," *Underline Science Inc.*, 2024. [Accessed by 25/07/2024].
25. P. Paramasivan, S. S. Rajest, K. Chinnusamy, R. Regin, and F. J. John Joseph, Eds., "Clinical and Comparative Research on Maternal Health," Advances in Medical Technologies and Clinical Practice, *IGI Global*, United States of America, 2024.