ABSTRACT

Lumpy Skin Disease (LSD) is a highly contagious viral disease that severely impacts cattle health, leading to substantial economic losses in the livestock industry through reduced productivity and increased mortality. Traditional diagnostic methods, often based on clinical observation and symptomatic analysis, are time-intensive and prone to subjective interpretation, which can delay critical intervention. This study presents an AI-enhanced approach to improve the accuracy and efficiency of LSD diagnosis by utilizing a hybrid model combining machine learning and image classification techniques. Leveraging a multi-modal dataset of clinical features and image data, the system integrates a tabular classification model for symptom analysis, an image classification model for visual lesion detection, and a Generative AI model to generate a comprehensive diagnostic report. The generated report delivers key insights, including the prediction outcome (LSD-positive or negative), critical observations, probable causes, and actionable recommendations for prevention and treatment. This approach offers a scalable and reliable diagnostic tool that enhances early intervention and supports effective disease management, ultimately aiming to reduce the spread of LSD, protect livestock health, and ensure agricultural productivity.

CHAPTER 1
INTRODUCTION

1.1 OBJECTIVE

Lumpy Skin Disease (LSD) is a viral infection that primarily affects cattle and is caused by the Capripoxvirus. The disease is characterized by the appearance of nodules on the skin, which can spread across the animal’s body, leading to severe health issues, including fever, emaciation, and reduced milk production. LSD is a vector-borne disease, primarily transmitted through biting insects such as mosquitoes and flies. Infected animals may suffer from secondary infections due to the open sores on their skin, which exacerbate the disease's impact. Mortality rates for LSD are generally low, but the economic impact is significant due to losses in milk production, decreased quality of hides, increased veterinary costs, and, in severe cases, culling of infected animals. Outbreaks of LSD can cause substantial disruptions in livestock-based economies, particularly in regions where cattle are a critical component of the agricultural sector.

Environmental and Socio-Economic Impact

The disease also poses a threat to the environment and local ecosystems. As farmers attempt to contain LSD, there may be increased use of chemical pesticides to control insect vectors, which can negatively impact biodiversity and water quality. Additionally, the movement restrictions imposed on livestock during outbreaks can have ripple effects, affecting meat and dairy supply chains, leading to higher prices, and impacting food security in affected regions. As a result, LSD represents not only an animal health issue but also a socio-economic and environmental challenge. The rapid and accurate diagnosis of LSD is essential to contain its spread and minimize its impact. However, traditional diagnostic methods, which rely on clinical examination and laboratory testing, may not always be feasible due to time constraints, limited resources, and the potential for subjective interpretation.

Role of Artificial Intelligence in Disease Diagnosis

Advances in artificial intelligence (AI) present promising solutions for enhancing disease detection and management in the veterinary field. AI-driven diagnostic tools have the potential to transform the way diseases like LSD are identified and monitored by providing rapid, accurate, and scalable solutions. Machine learning algorithms, when trained on relevant data, can identify patterns and anomalies in clinical and visual data, often with precision that rivals human expertise. Image-based AI models can detect visual signs of the disease, such as skin nodules, while tabular data models can analyze clinical indicators, providing a comprehensive diagnostic framework. AI systems can also be continually improved through exposure to new data, enhancing their effectiveness over time. This scalability and adaptability make AI a valuable asset in managing diseases that are prone to regional outbreaks and that evolve with environmental conditions.

Project Objective and Approach

The objective of this project is to develop an integrated AI-driven solution for the early and accurate detection of Lumpy Skin Disease. The system leverages a hybrid approach, combining tabular data analysis through machine learning, image classification for visual symptom detection, and a Generative AI model for synthesizing a detailed diagnostic report. The report generated by this system goes beyond simple classification; it includes critical insights into the disease's potential causes, key observations derived from the data, and actionable recommendations for treatment and prevention. This holistic approach enables veterinarians and livestock managers to make informed decisions quickly, facilitating early intervention and improving overall disease management.

Accessibility and Scalability of the Solution

In regions where veterinary resources are limited, this AI-based solution can be deployed to provide essential diagnostic support. The automated diagnostic tool is designed to be user-friendly and accessible, allowing field workers, farmers, and veterinarians to utilize the technology without extensive training. By integrating multiple data sources and advanced AI algorithms, the project aims to enhance LSD detection accuracy, minimize diagnostic delays, and reduce the reliance on traditional, resource-intensive diagnostic methods. Ultimately, this system aims to mitigate the socio-economic impact of LSD, reduce the strain on environmental resources, and improve the resilience of livestock-dependent communities.

Significance of the Project

This project exemplifies how AI can play a vital role in the agricultural sector, addressing critical challenges in animal health and sustainability. By enabling efficient and precise disease management, the technology not only contributes to animal welfare but also supports food security and economic stability in regions affected by LSD. The successful implementation of this AI-driven diagnostic tool could serve as a model for tackling other livestock diseases, paving the way for a more technologically empowered approach to veterinary medicine and livestock management.

CHAPTER 2
LITERATURE SURVEY

Lumpy Skin Disease Diagnosis in Cattle: A Deep Learning Approach Optimized with RMSProp and MobileNetV2

The study’s primary contribution lies in the integration of MobileNetV2, a lightweight and computationally efficient convolutional neural network (CNN) architecture, with the RMSprop optimization technique. MobileNetV2 was selected for its ability to reduce the computational load while maintaining performance, making it feasible for deployment on mobile or edge devices in rural settings. The architecture is particularly effective in image classification tasks where resource limitations are a concern, making it well-suited for applications in agriculture and veterinary medicine. MobileNetV2 utilizes depthwise separable convolutions, which effectively reduce the number of parameters and computational costs compared to traditional CNN architectures, without compromising the model’s ability to capture essential features in complex images, such as the lesions and skin textures seen in LSD-affected cattle.

The RMSprop optimizer was chosen to address the fading gradient problem commonly encountered in deep learning models, particularly in CNNs where gradient updates are crucial for convergence. RMSprop adaptively adjusts the learning rate, which helps stabilize the training process, reduces the likelihood of overshooting the optimal solution, and enables the model to converge more quickly. This optimization approach not only accelerates the training phase but also enhances the model's generalization capability, making it more reliable when classifying unseen images in a test set.

Dataset Preparation and Methodology

The dataset used in this study comprised 464 images of healthy cows and 329 images of cows with LSD. These images were further divided into training and testing sets to ensure a balanced representation. In the preprocessing stage, images were resized to 224x224 pixels, a resolution optimized for MobileNetV2, which allows for efficient processing while retaining sufficient detail for accurate classification. Augmentation techniques were applied to increase dataset variability and robustness, thereby reducing overfitting and improving the model’s performance on test data.

The training process involved feeding the images into the MobileNetV2 architecture, with the RMSprop optimizer adjusting the weights to minimize classification error. The model was trained over five epochs, with each epoch lasting approximately 30-40 seconds, demonstrating the efficiency of the proposed approach in terms of both time and resource consumption. The final accuracy achieved on the test set was 95%, which not only surpasses existing benchmarks for LSD diagnosis in cattle but also underscores the effectiveness of MobileNetV2 and RMSprop in handling this type of image classification problem.

Comparative Analysis and Model Performance

The performance of the MobileNetV2 model with RMSprop optimization was benchmarked against other deep learning models. Traditional CNN models, although effective, often require substantial computational power and memory, making them less suitable for field deployment. In contrast, MobileNetV2, with its reduced parameter count and lower memory usage, demonstrated a clear advantage in terms of computational efficiency. The proposed model's classification accuracy exceeded that of existing models by 4-10%, highlighting its superior ability to distinguish between healthy and LSD-affected cattle. This improvement in accuracy, coupled with faster inference times, positions the MobileNetV2-based approach as a practical solution for real-world diagnostics, where quick and accurate results are essential.

Challenges and Limitations

Despite the promising results, certain limitations were noted in the study. One major challenge lies in the resolution and quality of images used in training. Images taken from varying distances and under different lighting conditions can affect the model's ability to accurately detect LSD symptoms, particularly if lesions or nodules are subtle or partially obscured. Moreover, images with background noise, such as human figures or additional animals, posed classification challenges, occasionally leading to misclassification. Future research could address these issues by incorporating additional preprocessing steps, such as background removal or segmentation techniques, to isolate the cow from its surroundings, thus improving the focus on disease-relevant features.

Conclusion

The study on LSD diagnosis using MobileNetV2 and RMSprop highlights a significant advancement in the use of deep learning for veterinary applications. By achieving a classification accuracy of 95%, this approach demonstrates the effectiveness of lightweight CNN architectures in providing quick, reliable diagnostics, essential for managing contagious diseases in cattle. The combination of MobileNetV2's efficient design with the adaptability of RMSprop optimization offers a practical solution for field diagnostics, bridging the gap between AI advancements and real-world veterinary needs. Future work on this model has the potential to expand its applicability to a broader range of livestock diseases, positioning AI as a pivotal tool in the proactive management of animal health.

Applying Different Resampling Strategies in Random Forest Algorithm to Predict Lumpy Skin Disease

Lumpy Skin Disease (LSD) presents a significant challenge to livestock health, characterized by its impact on cattle through symptoms such as skin nodules, fever, and weight loss. The disease's rapid transmission through insect vectors like flies and ticks makes early diagnosis crucial to prevent outbreaks. However, accurately diagnosing LSD can be challenging due to the infrequency of positive cases in data, leading to class imbalance, a common problem in rare disease datasets. Class imbalance skews model performance toward the majority class, often at the expense of accurate minority class prediction. This study addresses the problem by employing resampling techniques to improve the performance of a Random Forest classifier in identifying LSD-positive cases, ensuring a more balanced and accurate prediction model.

Random Forest and Imbalanced Data

Random Forest, a widely used machine learning algorithm, combines multiple decision trees to improve classification accuracy and reduce overfitting. However, its performance suffers in datasets with imbalanced class distributions, as the majority class tends to dominate the predictions. In veterinary diagnostics, where minority classes (e.g., LSD-positive cases) are critical to detect, ignoring the class imbalance can lead to underdiagnosis and delayed intervention. The study thus explores resampling techniques, specifically Random Undersampling and Synthetic Minority Oversampling Technique (SMOTE), to balance the classes within the dataset before training the Random Forest model. By enhancing the algorithm's sensitivity to the minority class, these resampling strategies help ensure that the model accurately identifies LSD cases, which is vital for effective disease management and control.

Performance Comparison and Evaluation Metrics

The study evaluated the impact of these resampling techniques on the Random Forest classifier using metrics such as recall, precision, and F1 score. Precision measures the accuracy of positive predictions, while recall indicates the model's ability to identify all true positives, making it a vital metric in medical diagnostics where the priority is to minimize false negatives. The F1 score, as the harmonic mean of precision and recall, offers a balanced evaluation of the model's accuracy for both classes. Results indicated that SMOTE improved recall and F1 scores by 1-2% over Random Undersampling, suggesting that generating synthetic data is more effective than merely reducing majority instances. The improved recall and F1 scores highlight SMOTE's effectiveness in maintaining data diversity while enhancing the model's sensitivity to minority cases, thus addressing the primary challenge posed by imbalanced datasets in disease detection.

Challenges and Limitations

Despite the improvements achieved with resampling, certain limitations were noted in the study. One challenge lies in the reliance on a single classifier, Random Forest, which, although effective, may not be optimal for all aspects of LSD detection. Exploring other algorithms or ensemble methods could provide further improvements in model performance. Another limitation involves the nature of synthetic data generation with SMOTE. While SMOTE helps balance the dataset, it may also introduce noisy data points if synthetic samples do not closely resemble real cases, potentially leading to overfitting. Ensuring the quality of synthetic data is therefore crucial, as inaccurate representations could reduce model generalization on actual test cases. The study acknowledges these limitations and suggests that combining SMOTE with additional data augmentation techniques could further improve model robustness and reduce the likelihood of overfitting.

Implications for Veterinary Diagnostics

This study's findings underscore the importance of handling data imbalance in veterinary diagnostics, where minority cases often represent the more clinically significant outcomes. By demonstrating that SMOTE enhances recall and F1 scores, this research highlights how resampling can make machine learning models more reliable for rare disease detection. In the context of LSD, where timely identification can prevent widespread outbreaks, a model that accurately detects positive cases is essential for effective disease control. The study thus offers valuable insights for future research in animal health diagnostics, suggesting that data balancing techniques are critical for achieving high-performance models, especially in veterinary contexts where obtaining large, balanced datasets is challenging.

Conclusion

This study's approach to addressing class imbalance in LSD prediction through resampling techniques offers a valuable advancement in the field of veterinary diagnostics. By improving the Random Forest classifier's recall and F1 score, particularly through the use of SMOTE, the research demonstrates that data balancing is essential for achieving reliable disease detection. The findings emphasize that resampling methods, when carefully implemented, can significantly enhance model performance in rare disease scenarios. This approach could serve as a model for other veterinary applications where data imbalance poses a challenge, making machine learning tools more effective for animal health management. Future research can build on these results by exploring ensemble methods, multi-modal data integration, and improved data augmentation, advancing the field of AI-driven veterinary diagnostics.

CHAPTER 3
SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing systems for diagnosing Lumpy Skin Disease (LSD) in cattle rely on visual inspections, manual assessments, and traditional machine learning models. While effective, these approaches are often time-consuming and lack adaptability to diverse field conditions, limiting their responsiveness in real-time disease management.

3.1.1 DRAWBACKS

While useful, existing LSD diagnostic methods have critical limitations, including time-intensive processes and limited accuracy in handling imbalanced data. These constraints hinder timely, accurate disease detection, especially in remote settings.

- **Time-Consuming and Resource-Intensive:** Traditional methods often involve physical examinations or laboratory testing, which can be time-consuming and require specialized equipment and expertise. This slows down the diagnostic process, making early detection challenging, especially in remote areas with limited resources.

- **Limited Accuracy in Imbalanced Data:** Machine learning models traditionally applied in this field frequently struggle with class imbalance, as healthy cattle images significantly outnumber images of infected cattle. This imbalance can lead to biased predictions, with models favoring the majority class and thus reducing accuracy in identifying actual disease cases.

- **High Computational Requirements:** Older deep learning models applied to LSD diagnosis often require extensive computational power and memory, making them difficult to implement on mobile or edge devices. This restricts their use in field settings where quick, portable diagnostics are needed.

- **Dependence on Traditional Data Processing:** Most existing methods focus on either image or tabular data independently, without integrating multiple data sources. This approach may overlook important indicators that a combined analysis could reveal, reducing diagnostic robustness.

- **Delayed Intervention and Disease Control:** Since traditional methods may delay diagnosis, there is often a lag in response, which can lead to uncontrolled disease spread, affecting entire herds and increasing economic loss.

3.2 PROPOSED SYSTEM

The proposed system utilizes a hybrid AI-driven approach to enhance the accuracy and accessibility of Lumpy Skin Disease (LSD) diagnosis in cattle, with Generative AI as a key component. By integrating image classification through MobileNetV2 with Random Forest, enhanced by resampling techniques, the system addresses class imbalance, computational efficiency, and diagnostic accuracy. Generative AI plays a central role by synthesizing comprehensive diagnostic reports based on input images and clinical data, providing veterinarians with insights into key symptoms, probable causes, and recommended interventions. This system, optimized for mobile and edge devices, enables rapid, field-ready diagnostics, essential for timely intervention in remote or resource-limited environments. This generative approach not only overcomes traditional limitations but also enhances interpretability, empowering users with actionable insights for effective disease management.

3.2.1 SYSTEM REQUIREMENTS

The proposed system for diagnosing Lumpy Skin Disease (LSD) is designed to meet both functional and non-functional requirements to ensure reliable, efficient, and user-friendly performance. Functionally, the system must be capable of processing and classifying cattle images, generating detailed diagnostic reports through Generative AI, and handling clinical data integration for comprehensive analysis. Non-functional requirements include performance standards, ensuring quick processing times and high diagnostic accuracy, as well as scalability, usability, and data security. These requirements collectively support the system’s goal of providing accurate, accessible, and timely diagnostics in diverse and resource-limited environments.

3.2.2.1 FUNCTIONAL REQUIREMENTS

Image Processing and Classification

The system's primary function is to analyze and classify images of cattle to determine whether they are affected by Lumpy Skin Disease (LSD) or healthy. Using MobileNetV2, a lightweight and efficient deep learning model, the system processes high-resolution images, extracting relevant features to identify disease symptoms such as skin lesions and nodules. This function is essential for accurately detecting visual indicators of LSD in cattle, supporting timely and precise diagnostics. The model’s performance must be optimized to handle a wide range of image qualities and lighting conditions typically encountered in field environments.

Generative AI for Diagnostic Reporting

A core feature of the system is the integration of Generative AI, which synthesizes comprehensive diagnostic reports based on the analysis of images and input data. These reports include key findings, possible causes, and recommended actions, providing veterinarians and field workers with actionable insights. The Generative AI component is designed to interpret and compile information from both image data and clinical input, ensuring that the final report is clear, informative, and easily understandable. This functionality not only enhances the diagnostic process but also supports decision-making in disease management.

Clinical Data Processing and Integration

In addition to image analysis, the system must process clinical data—such as symptoms, health history, and environmental factors—that may impact the diagnosis of LSD. By integrating clinical data with image analysis, the system provides a more comprehensive assessment of the animal's health. This multi-modal data processing allows the model to consider various factors that might not be visually apparent, thereby increasing diagnostic accuracy. This functionality is particularly beneficial for cases where visual symptoms are ambiguous or less pronounced, as it allows for a holistic diagnostic approach.

User Interface for Diagnostics and Feedback

The system includes a user interface that allows veterinarians, field workers, and other users to interact with the diagnostic tool. This interface provides an accessible platform for users to upload images, input clinical data, and receive diagnostic reports generated by the system. Additionally, the interface should allow users to provide feedback on diagnostic accuracy and report clarity, which can be used to refine the model over time. Ensuring a user-friendly interface is crucial, as it promotes ease of use and ensures that the system can be operated effectively even by users with minimal technical expertise.

3.2.1.2 NON-FUNCTIONAL REQUIREMENTS

Performance and Efficiency

The system must operate efficiently, with quick processing times to ensure timely diagnosis in field environments. The image classification and diagnostic report generation should be completed within a few seconds to enable prompt decision-making. High-performance standards are essential, as slow processing could delay critical intervention, especially during outbreaks. The system is optimized for lightweight architecture (using MobileNetV2) to achieve rapid analysis without requiring extensive computational resources, making it suitable for mobile and edge devices commonly used in rural and resource-limited settings.

Scalability and Flexibility

The system must be scalable to handle an increasing number of users and diagnostic requests without compromising performance. This requirement ensures that the system can support larger deployments, such as in areas with high cattle populations or in situations where multiple users need simultaneous access. Additionally, the system should be flexible enough to adapt to varying data inputs, including different image qualities, formats, and clinical data parameters. This adaptability is crucial for deploying the system across diverse environments and use cases, from small farms to large-scale livestock operations.

Reliability and Accuracy

Reliability is a critical non-functional requirement, as the system must provide consistently accurate diagnoses with minimal error rates. This involves thorough testing and validation to ensure that the model performs accurately across different conditions, such as varied lighting, animal breeds, and environmental backgrounds. A reliable system builds user confidence, particularly in remote and rural areas where access to veterinary resources may be limited. Accuracy in predictions and report generation is paramount, as misdiagnoses could lead to improper treatment or lack of intervention, potentially causing further disease spread.

Usability and User Accessibility

The system must be user-friendly and accessible to individuals with varying levels of technical expertise, including farmers, veterinarians, and field workers. A simple, intuitive interface allows users to navigate the system easily, upload images, and interpret reports without extensive training. Usability considerations also include providing clear and concise diagnostic reports, ensuring that essential information is readily understandable and actionable. This requirement is particularly important for enhancing the system's adoption in low-resource settings, where users may not have technical backgrounds.

Security and Data Privacy

Given that the system will handle sensitive diagnostic and clinical data, it is essential to ensure robust security and data privacy measures. All data transactions, including image uploads and diagnostic results, must be encrypted to protect user information and maintain confidentiality. Access control mechanisms should be implemented to prevent unauthorized access to sensitive data. Compliance with data privacy regulations is also necessary to safeguard user information, which is especially critical in healthcare-related applications. Ensuring data security fosters trust in the system and encourages its use by providing reassurance that user data will be handled responsibly.

CHAPTER 4
SYSTEM SPECIFICATION

4.1 HARDWARE SPECIFICATIONS

The hardware specifications for the proposed system are designed to support efficient image processing, Generative AI-based reporting, and real-time diagnostics in resource-limited environments. To achieve these goals, the system requires certain minimum and recommended hardware components, especially for deployments in rural and mobile settings.

Minimum Requirements

For basic functionality, the system can operate on devices with modest specifications. This includes:

- **CPU:** At least a quad-core processor (e.g., Intel Core i5 or equivalent) to handle data processing tasks.
- **RAM:** A minimum of 8 GB to support smooth operation of the image processing and machine learning models.
- **Storage:** 128 GB of storage space to accommodate the diagnostic application, data storage, and caching needs.
- **GPU:** For edge or mobile devices, a basic integrated GPU is sufficient for lightweight processing, although it may limit model complexity and speed.

Recommended Requirements

To optimize performance and allow for faster processing, the following recommended specifications are advised:

- **CPU:** Octa-core processor (e.g., Intel Core i7 or equivalent) for enhanced data processing speed and multitasking capabilities.
- **RAM:** 16 GB or more to support the efficient handling of larger datasets and real-time model inference.
- **Storage:** 256 GB SSD to ensure quick access to stored data and to provide sufficient space for storing images, clinical data, and cached results.
- **Dedicated GPU:** A dedicated GPU (e.g., NVIDIA GTX 1050 or higher) for systems running more complex AI models, ensuring faster image analysis and report generation times.

Edge Device Requirements

For remote or field applications, portable and edge devices can be utilized:

- **Mobile Device/Tablet:** Devices with mobile processors and integrated GPUs capable of running lightweight versions of the diagnostic model (e.g., ARM Cortex-A processors).

- **Battery Life:** Sufficient battery capacity for continuous usage in field conditions, ideally supporting several hours of operation.
- **Connectivity:** Wireless connectivity options (Wi-Fi, Bluetooth) for data transfer to centralized systems or cloud storage when necessary.

Cloud/Server Deployment

For larger-scale implementations, the system can also be deployed on cloud-based servers to handle intensive processing tasks:

- **CPU:** Multi-core server-grade processors (e.g., Intel Xeon or AMD EPYC).
- **RAM:** 32 GB or more for handling large volumes of data and concurrent requests.
- **High-Performance GPU:** NVIDIA Tesla or A100 series GPUs for accelerated machine learning model inference.
- **Storage:** 1 TB or more for data storage and model caching, especially if handling data from multiple sources.

These hardware specifications ensure the system's adaptability across different environments, from mobile deployments in remote areas to cloud-based operations for large-scale diagnostics.

4.2 SOFTWARE SPECIFICATIONS

The software specifications for the proposed system are designed to ensure compatibility, efficiency, and ease of deployment across various platforms. These specifications include essential software tools, frameworks, and libraries required to run the image classification, Generative AI reporting, and data processing components.

Operating System

The system is designed to be compatible with multiple operating systems to facilitate flexibility in deployment:

- **Linux (Ubuntu 18.04 or later):** Preferred for server and cloud-based deployments due to its stability, security, and compatibility with machine learning libraries.
- **Windows 10 or later:** Suitable for desktop and local deployment, especially for ease of use in non-technical environments.
- **Android/iOS:** For mobile and edge deployments, enabling diagnostic capabilities on portable devices.

Machine Learning Frameworks

The system relies on advanced machine learning frameworks to support image processing, classification, and Generative AI functionalities:

- **TensorFlow 2.x or PyTorch:** These frameworks are used to build and deploy the image classification model (MobileNetV2) and the Generative AI component. Both frameworks support GPU acceleration, which improves processing speed and efficiency.

- **Keras:** Integrated with TensorFlow, Keras provides a user-friendly interface for building and fine-tuning deep learning models, simplifying the development and deployment process.
- **Scikit-Learn:** Used for implementing machine learning algorithms like Random Forest and for data preprocessing tasks, providing essential tools for model training, evaluation, and resampling techniques such as SMOTE.

Data Processing Libraries

The system requires robust data processing libraries to handle image data and clinical information:

- **NumPy and Pandas:** Essential for data manipulation, cleaning, and preprocessing, enabling smooth handling of structured data.
- **OpenCV:** Used for image processing tasks, including resizing, normalization, and other preprocessing steps needed to optimize images for the classification model.

Generative AI Tools

The system uses advanced Generative AI tools to produce comprehensive diagnostic reports from multimodal data, enhancing interpretability and usability.

- **Gemini API (Gemini-1.5-Flash-Latest):** The primary model currently used is Gemini-1.5, a proprietary, high-performance multimodal model accessed via the Gemini API. This model is capable of processing both image and text data, allowing it to generate detailed, contextually relevant diagnostic reports based on image analysis and clinical information.
- **LLama 3 (Open-Source Alternative):** For larger-scale deployments, open-source models like LLama 3 may be used with fine-tuning to adapt the model specifically to LSD diagnostics. LLama 3 offers flexibility and cost-efficiency, enabling customized solutions while maintaining high performance.

4.2.1 FRONTEND

The frontend of the system is designed to provide a user-friendly, accessible interface for interacting with the diagnostic tool, supporting both web and mobile users. Currently, Gradio is used for deployment, offering an efficient and quick setup for displaying the diagnostic functionalities in a streamlined web interface. Gradio enables easy data input, image uploading, and real-time interaction, making it ideal for initial deployment and rapid testing.

In future upgrades, the frontend will be enhanced to offer a more robust web and mobile experience:

- **Framework and Architecture:** The system will transition to a more scalable web application, utilizing FastAPI as the backend framework to handle data processing and API requests efficiently. For the frontend, React.js will be implemented to provide a dynamic and responsive interface for web users, while React Native will be used to support mobile app development for Android and iOS. This combination allows for a unified codebase, ensuring consistency across web and mobile platforms.
- **User Input and Image Upload:** The frontend will retain user-friendly features for uploading images and entering clinical data. React-based components will allow users to drag-and-drop images or browse files for upload, along with structured input fields for entering relevant data. These features will be optimized for seamless interaction across devices, ensuring accessibility in field environments.
- **Feedback Collection:** A feedback mechanism will be incorporated, enabling users to rate the accuracy and clarity of the diagnostic reports. This feedback will be stored and used to further fine-tune the AI models, enhancing the system's effectiveness over time.

4.2.2 BACKEND & DATABASE

The backend and database components of the system are designed to support efficient data processing, secure storage, and seamless communication between the frontend and core application. The core app, built in Python, handles image processing, machine learning model inference, and Generative AI report generation, providing the system's primary diagnostic capabilities.

- **Backend Framework:** Utilizes FastAPI for high-performance, asynchronous API handling in Python. Enables fast data processing and real-time communication between frontend and core diagnostic functions.
- **Database:** MySQL is used for secure storage and management of diagnostic data, user information, and image metadata. Offers reliability and scalability for structured data, supporting efficient data retrieval.
- **Data Security:** Implements encryption for sensitive data to ensure privacy and compliance with data protection standards.
- **Scalability:** FastAPI and MySQL provide a flexible, scalable solution that supports future system expansion.

CHAPTER 5
PROPOSED SYSTEM

5.1 ARCHITECTURE OF LSD PREDICTION

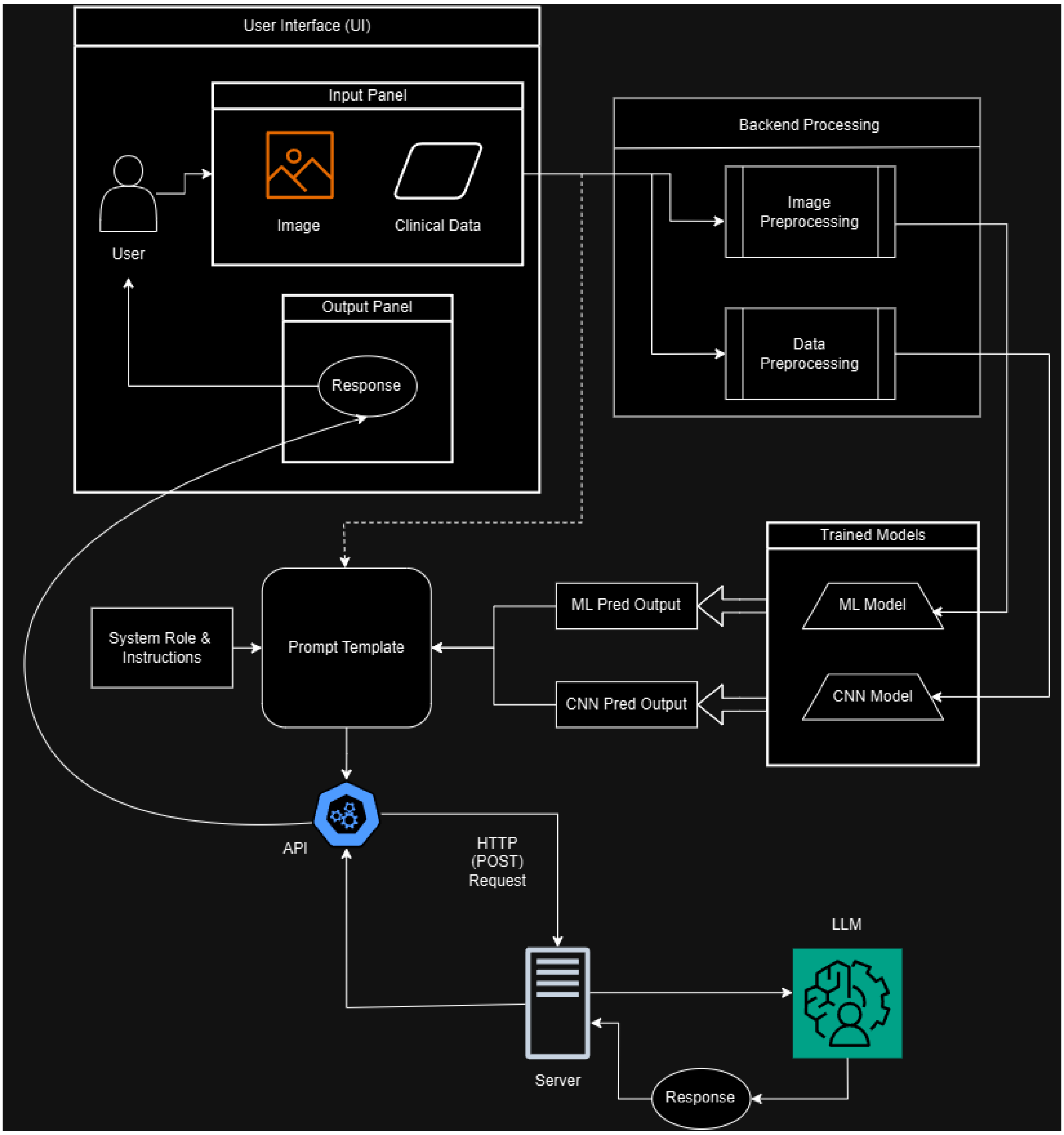

Fig 5.1 Architecture Diagram of LSD Prediction

5.1.1 DATA COLLECTION

The data collection process for this system involved gathering two primary datasets: one for the machine learning (ML) model and another for the Convolutional Neural Network (CNN) model. These datasets provide the foundation for accurately diagnosing Lumpy Skin Disease (LSD) and differentiating it from normal skin conditions in cattle. The ML model utilizes a structured dataset sourced from Kaggle, while the CNN model leverages an image dataset obtained from the Mendeley Data platform. Together, these datasets enable a comprehensive, multi-modal approach to LSD diagnosis, covering both numerical and visual inputs.

The structured dataset for the ML model was collected from Kaggle, comprising 24,803 rows and 20 columns. This dataset includes various clinical and environmental indicators relevant to diagnosing LSD. After careful selection, ten input features were chosen to serve as predictive variables for the model, with one column designated as the output target, indicating the presence or absence of LSD. The extensive size of this dataset allows the ML model to learn a wide range of conditions and factors associated with the disease, supporting robust and reliable predictions.

The image dataset for the CNN model was sourced from Mendeley Data, a platform providing high-quality research datasets. This dataset is structured into two main subfolders, labeled "Lumpy Skin" and "Normal Skin," with a total of 1,024 images. Specifically, the "Lumpy Skin" folder contains 324 images, while the "Normal Skin" folder includes 700 images. This balanced structure allows the CNN model to accurately differentiate between LSD-affected skin and healthy skin, enhancing the system's diagnostic accuracy by using visual data to complement the clinical data.

Each image in the dataset has been preprocessed and standardized to ensure consistency, resized to 256x256 pixels in PNG format. The pre-organized structure and preprocessing of the images make this dataset highly suitable for training CNN models, as it minimizes variation in image quality and resolution. By combining these structured and image datasets, the system can integrate clinical data analysis with visual assessment, offering a more comprehensive and effective approach to detecting and diagnosing Lumpy Skin Disease in cattle.

	x	y	region	country	reportingDate	cld	dtr	frs	pet	pre	tmn	tmp	tmx	vap	wet	elevation	dominant_land_cover	X5_Ct_2(
0	90.380931	22.437184	Asia	Bangladesh	10/9/2020	41.6	12.8	0.00	2.3	1.7	12.7	19.1	25.5	15.7	0.00	147	2	27970.
1	87.854975	22.986757	Asia	India	20/12/2019	40.5	13.3	0.00	2.4	0.0	13.2	19.8	26.5	16.3	0.00	145	2	25063.
2	85.279935	23.610181	Asia	India	20/12/2019	27.3	13.6	0.08	2.3	0.6	9.4	16.2	23.0	13.0	0.98	158	2	6038.
3	81.564510	43.882221	Asia	China	25/10/2019	45.3	12.8	31.00	0.4	8.8	-22.5	-16.1	-9.7	0.9	4.64	178	2	760.
4	81.161057	43.834976	Asia	China	25/10/2019	38.8	13.2	31.00	0.4	10.5	-20.4	-13.8	-7.2	1.2	1.69	185	3	270.
5	81.248335	43.966008	Asia	China	25/10/2019	38.8	13.2	31.00	0.4	10.5	-20.4	-13.8	-7.2	1.2	1.69	185	3	270.
6	81.074165	43.836019	Asia	China	25/10/2019	38.8	13.2	31.00	0.4	10.5	-20.4	-13.8	-7.2	1.2	1.69	185	3	270.
7	81.547132	43.688309	Asia	China	25/10/2019	45.3	12.8	31.00	0.4	8.8	-22.5	-16.1	-9.7	0.9	4.64	178	2	760.
8	81.239566	43.591386	Asia	China	25/10/2019	38.8	13.2	31.00	0.4	10.5	-20.4	-13.8	-7.2	1.2	1.69	185	3	270.
9	81.324148	43.978094	Asia	China	25/10/2019	38.8	13.2	31.00	0.4	10.5	-20.4	-13.8	-7.2	1.2	1.69	185	3	270.

Fig 5.1.1.1 Clinical Dataset

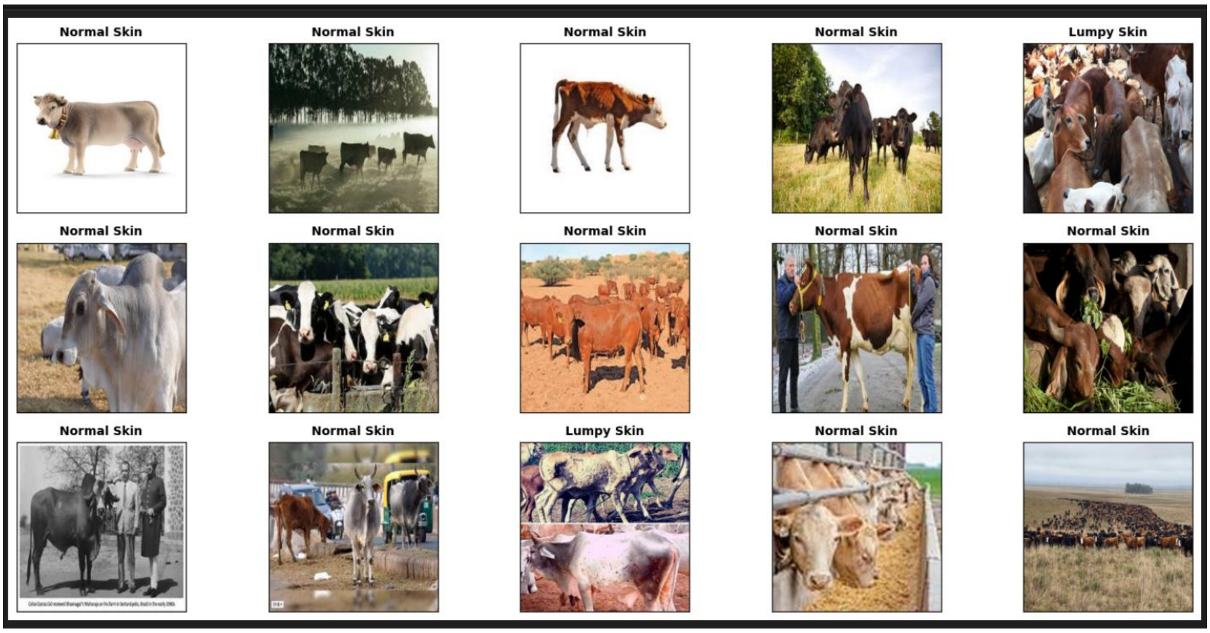

Fig 5.1.1.2 Image Dataset

5.1.2 DATA PREPROCESSING

Data preprocessing is a crucial step in preparing the datasets for the ML and CNN models, ensuring high-quality input data for accurate and consistent predictions. For the structured dataset from Kaggle, the preprocessing process involved several steps to handle missing values, identify and address outliers, and standardize the data formats. Initially, any rows with significant missing values were carefully examined; where feasible, missing entries were filled using statistical methods such as mean or median imputation to retain as much information as possible without compromising data integrity.

Next, outlier detection techniques were applied to identify any data points that deviated significantly from typical values. Outliers can negatively impact model performance by skewing the training process, so they were either removed or capped based on statistical thresholds. After outlier handling, the dataset was standardized by converting all features to consistent units and formats. This step ensures that each feature has a similar scale, which helps improve the convergence speed of the ML model during training. Finally, the dataset was reduced to 10 input features and 1 output target, optimizing it for training while retaining critical predictive variables.

For the image dataset from Mendeley Data, preprocessing focused on ensuring uniformity and optimizing the images for CNN model input. Each image was resized to a standard 256x256 resolution in PNG format to maintain consistency across all samples, as CNN models require fixed input dimensions. Additionally, color normalization techniques were applied to adjust brightness and contrast levels, creating a uniform appearance across all images. This step helps the CNN model focus on identifying key features related to LSD symptoms rather than being affected by lighting or color variations.

Data augmentation techniques, such as horizontal flipping, rotation, and zoom adjustments, were also implemented on the image dataset to enhance model generalization and prevent overfitting. By generating slightly varied versions of the images, the CNN model gains exposure to a wider range of scenarios, improving its ability to accurately classify images in diverse real-world conditions. Through these preprocessing steps, both the structured and image datasets were prepared to maximize model performance, ensuring a robust and reliable diagnostic system for Lumpy Skin Disease.

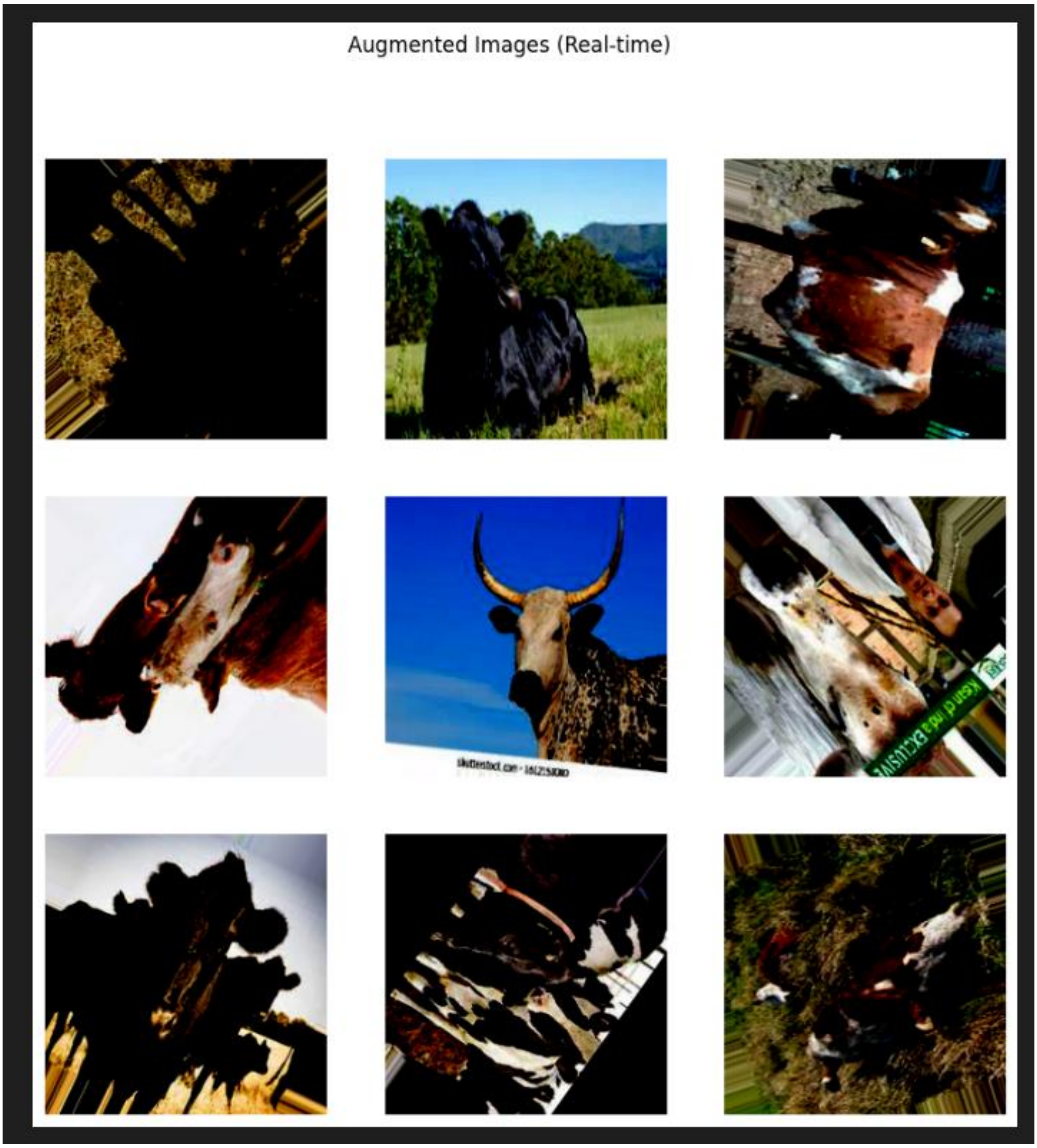

Fig 5.1.2.1 Data Augmentation for CNN

5.1.3 FEATURE EXTRACTION

Feature extraction is a key process in identifying and selecting the most relevant variables from the dataset to train the ML model for Lumpy Skin Disease (LSD) diagnosis. Out of the original 20 columns in the Kaggle dataset, a refined set of 10 features was selected based on their predictive relevance to LSD occurrence. These features were carefully chosen to include environmental and climatic factors that are likely to influence disease transmission and manifestation, as well as geographic variables that provide contextual information about the disease's spread.

The following features were selected for model training:

- **Longitude and Latitude:** These geographic coordinates provide spatial information, helping the model account for location-specific variations in disease prevalence. These variables allow the model to consider regional patterns, such as areas where LSD outbreaks are more common due to environmental or ecological factors.

- **Monthly Cloud Cover:** This feature reflects the average cloudiness in a specific location, which can affect temperature and humidity levels, potentially influencing disease spread by creating favorable conditions for the vectors that transmit LSD.

- **Potential EvapoTranspiration:** This metric estimates the amount of moisture loss in the atmosphere, which indirectly affects environmental dryness and humidity. By including this feature, the model can account for regions with higher or lower moisture levels, conditions that may impact the transmission and survival of disease-carrying insects.

- **Precipitation:** Rainfall data is essential in understanding moisture availability, which affects vegetation growth and, consequently, cattle health. Increased precipitation may also contribute to breeding environments for vectors that spread LSD, making it an influential factor in predicting disease risk.

- **Minimum Temperature, Mean Temperature, and Maximum Temperature:** These temperature-related features provide a comprehensive view of climate conditions in the target area. Temperature influences the survival and activity of the vectors responsible for transmitting LSD, making these metrics valuable for the model in assessing disease risks in different climatic zones.

- **Vapour Pressure:** This feature is associated with atmospheric humidity, which can impact both vector behavior and cattle susceptibility to disease. High humidity, for example, often correlates with increased insect activity, which can elevate the likelihood of disease transmission.

- **Wet Day Frequency:** This metric indicates the number of days with measurable precipitation, offering additional insights into moisture levels and environmental conditions conducive to vector activity.

- **Lumpy** (Target Variable): This is the target variable indicating the presence or absence of LSD in each observation, enabling the model to learn patterns associated with disease occurrence based on the selected features.

By focusing on these 10 extracted features, the model is trained to recognize environmental and geographic factors that correlate with LSD outbreaks, allowing for accurate and context-aware predictions. This carefully curated feature set enhances the model's ability to identify key patterns in the data, supporting effective and reliable disease diagnostics.

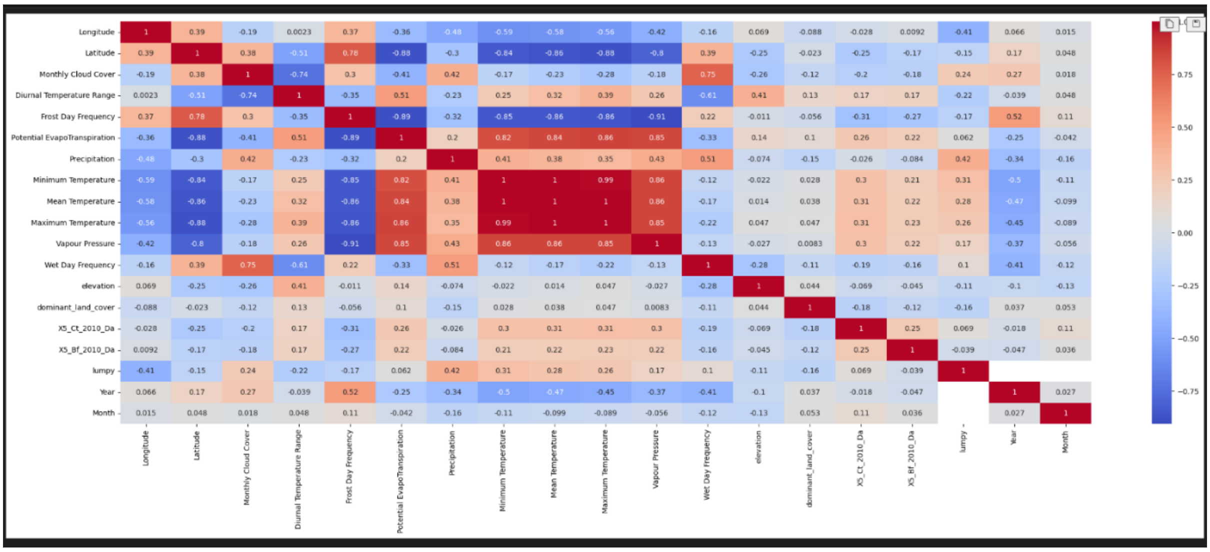

Fig 5.1.3.1 Correlation Analysis

5.1.4 MODEL TRAINING

The system's model training process involves both traditional machine learning (ML) models and a convolutional neural network (CNN) for comprehensive Lumpy Skin Disease (LSD) diagnostics. The ML models are trained on structured clinical and environmental data, while the CNN model is trained on images of cattle skin to distinguish between LSD-affected and normal skin.

ML Model Development

The development of machine learning (ML) models for Lumpy Skin Disease (LSD) prediction involved a systematic approach encompassing data preparation, model selection, hyperparameter tuning, and performance evaluation. By testing several algorithms and refining each through parameter tuning, we aimed to identify the best-performing model for accurately predicting LSD using clinical and environmental data. The dataset, derived from Kaggle, included structured features relevant to LSD, such as temperature, precipitation, and humidity, which are known to influence the transmission of the disease. After thorough data preprocessing, including handling missing values, standardizing features, and balancing classes, the data was split into training and testing sets to evaluate model generalization.

Model Selection and Training

We experimented with multiple ML algorithms to ensure a comprehensive comparison and select the most effective approach. The chosen models included K-Nearest Neighbors (KNN), Logistic Regression, Random Forest Classifier, Support Vector Classifier (SVC), XGBoost Classifier, and a Deep Learning model (Artificial Neural Network - ANN). These models were selected based on their diverse learning strategies and ability to handle structured data effectively.

- K-Nearest Neighbors (KNN): KNN is a distance-based algorithm that predicts the class of a sample by evaluating the classes of its closest neighbors. This model is simple yet effective for datasets with complex decision boundaries, especially when the classes are not linearly separable.

- Logistic Regression: Logistic regression is a statistical method used to predict binary outcomes. It provides probabilities for class membership, which is particularly useful for understanding model confidence. Its interpretability and simplicity make it a standard choice for binary classification problems.

- Random Forest Classifier: Random Forest is an ensemble learning method that builds multiple decision trees during training and averages their results. Known for its robustness, Random Forest is effective in handling high-dimensional data, making it suitable for this LSD prediction task where various environmental factors interact.

- Support Vector Classifier (SVC): SVC is a powerful classifier that attempts to find the hyperplane that best separates the classes in feature space. Its ability to use different kernels, such as linear and radial basis functions, allows it to capture complex relationships between features.

- XGBoost Classifier: XGBoost is an optimized gradient-boosting algorithm that builds a series of weak models sequentially, correcting errors from previous models. It’s particularly effective for structured data and is known for its speed and efficiency.

- Deep Learning (ANN): The ANN model was constructed with multiple layers to capture complex patterns in the dataset. Deep learning models can learn hierarchical feature representations, making them suitable for data with subtle, non-linear relationships.

Standardization and Splitting the Data

To prepare the data for model training, all features were standardized using StandardScaler. Standardization ensures that each feature has a mean of zero and a standard deviation of one, which is essential for algorithms sensitive to feature scaling, such as SVC and KNN. The data was then split into training and testing sets with an 80:20 ratio, preserving a portion of data for model evaluation. This separation allowed us to assess the model's ability to generalize to unseen data, an essential factor for real-world diagnostic applications.

Hyperparameter Tuning Strategies

To maximize model performance, each model underwent hyperparameter tuning using RandomizedSearchCV. RandomizedSearchCV explores a specified number of random combinations of hyperparameters, allowing us to find optimal parameters efficiently without exhaustive grid search. Different hyperparameter ranges were defined based on each model's characteristics:

- K-Nearest Neighbors: The hyperparameters tuned included the number of neighbors (n_neighbors), distance weights (weights), and distance metric (p).

- Logistic Regression: For logistic regression, penalties (l1, l2), solvers (liblinear, saga), and regularization strength (C) were optimized to control model complexity and improve generalization.

- Random Forest: We adjusted the number of trees (n_estimators), tree depth (max_depth), minimum samples for split (min_samples_split), and class weights (class_weight) to balance between bias and variance.

- SVC: For SVC, we tuned the regularization parameter (C), kernel type (linear, rbf, poly), and kernel coefficient (gamma) to control the model's decision boundary flexibility.

- XGBoost: In XGBoost, parameters such as the number of estimators (n_estimators), learning rate (learning_rate), and subsample ratio were fine-tuned to optimize model convergence.

- Deep Learning (ANN): The ANN model's hyperparameters, including the number of neurons in each layer, dropout rates, and learning rate, were optimized using Keras Tuner. By performing a hyperparameter search, we identified the best configuration to maximize validation accuracy.

Model Evaluation Results

After training and tuning, each model was evaluated based on several performance metrics: accuracy, precision, recall, and F1-score. These metrics provide a comprehensive view of each model’s effectiveness:

- K-Nearest Neighbors (KNN): The KNN model achieved an accuracy of 0.97, with a high precision of 0.91 and an F1-score of 0.91. Its strong performance indicates its ability to capture patterns in the dataset, although its reliance on distance metrics makes it sensitive to feature scaling.

- Logistic Regression: Logistic regression achieved an accuracy of 0.95, with lower recall at 0.75. While logistic regression is simple and interpretable, it struggled to capture complex patterns, leading to missed positive cases in some instances.

- Random Forest Classifier: The Random Forest model outperformed other models with an accuracy of 0.98, high precision (0.91), recall (0.90), and an F1-score of 0.90. Its ensemble structure allows it to generalize well, handling the diverse features and interactions in the dataset effectively.

- Support Vector Classifier (SVC): The SVC model demonstrated competitive performance with an accuracy of 0.96 and balanced metrics across precision, recall, and F1-score. Its flexibility with kernels allows it to perform well on non-linear relationships, making it a reliable choice.

- XGBoost Classifier: XGBoost achieved an accuracy of 0.97 with high recall (0.88), highlighting its strength in correctly identifying positive cases. Its robust gradient-boosting algorithm helped in capturing subtle interactions, making it effective for this structured dataset.

- Deep Learning (ANN): The ANN model reached an accuracy of 0.96, with a recall of 0.81. While it showed good performance, further tuning could potentially enhance its sensitivity to the minority class.

```
Performance of RandomForest on Test Data:
[[4285   69]
 [  55  552]]
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      4354
           1       0.89      0.91      0.90       607

    accuracy                           0.98      4961
   macro avg       0.94      0.95      0.94      4961
weighted avg       0.98      0.98      0.98      4961
...
    accuracy                           0.97      4961
   macro avg       0.94      0.93      0.93      4961
weighted avg       0.97      0.97      0.97      4961
```

Fig 5.1.4.1 Classification Report for Random Forest Model

Final Model Selection: Random Forest

Among the tested models, Random Forest emerged as the top performer with the highest accuracy of 0.98 and balanced precision, recall, and F1-score. Random Forest's ensemble approach, which aggregates predictions from multiple decision trees, enables it to capture complex feature interactions and avoid overfitting. Additionally, its robustness to outliers and class imbalance made it ideal for the LSD dataset, where environmental factors and disease spread are influenced by diverse variables.

The Random Forest model's ability to provide reliable predictions with high accuracy and balanced sensitivity and specificity makes it suitable for deployment in the LSD diagnostic tool. It can effectively process structured clinical and environmental data to predict disease occurrence, offering veterinarians a dependable tool for early intervention.

In conclusion, the systematic development and evaluation of multiple ML models revealed that the Random Forest classifier, with its ensemble-based learning and resilience to diverse input features, is best suited for this application. This model will be deployed as part of the LSD diagnostic system, providing accurate and actionable predictions to support effective disease management.

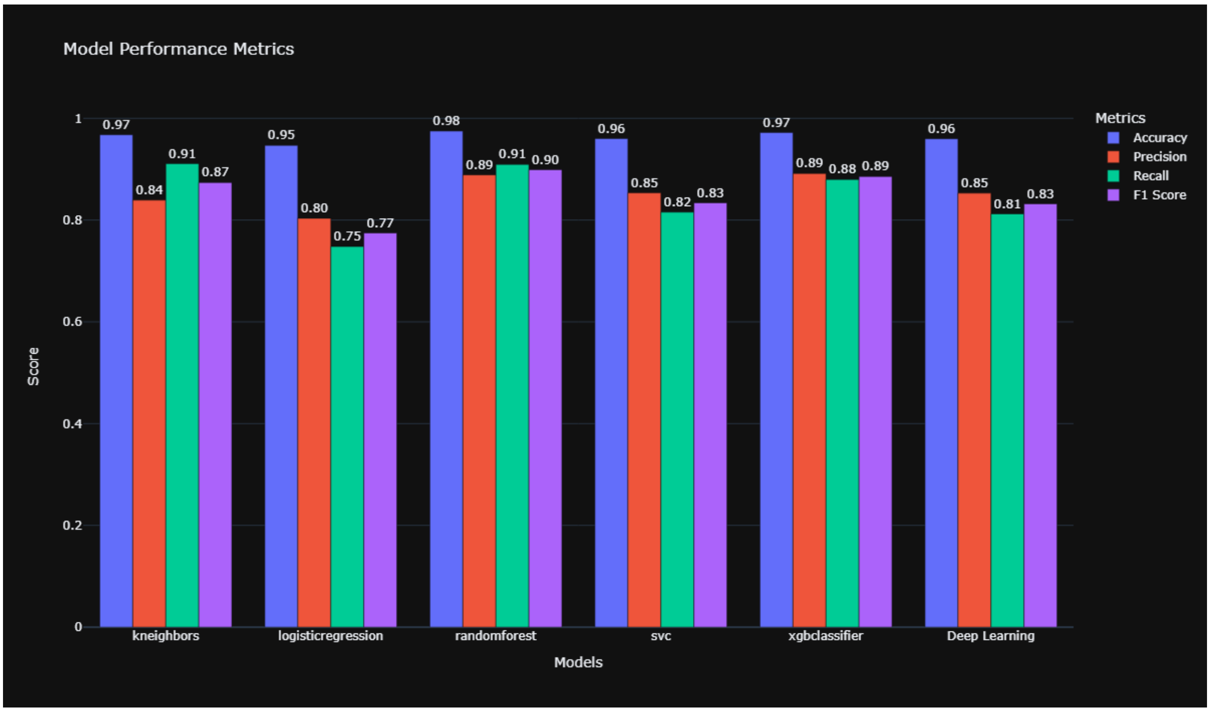

Fig 5.1.4.2 ML Model Performance Metrics

CNN Model Training

The Convolutional Neural Network (CNN) model for Lumpy Skin Disease (LSD) detection was developed to differentiate between images of cattle with lumpy skin lesions and those with healthy skin. The CNN model was built upon the pre-trained MobileNetV2 architecture, which is widely used for image classification tasks due to its efficiency and high performance on limited-resource devices. The model was fine-tuned to specifically identify

LSD in cattle using a dataset of images that were pre-categorized into two classes: “Lumpy Skin” and “Normal Skin.” This section discusses the data preparation, image augmentation, model architecture, training process, and evaluation results for the CNN model.

Data Preparation

To further streamline the training process, the dataset was split into training and testing sets, with 90% of the images allocated for training and 10% for testing. Additionally, within the training set, a subset was reserved for validation to monitor the model’s performance and adjust weights accordingly. This division was done using the train_test_split function to ensure a balanced and representative split of data, critical for accurate model evaluation and generalization.

Image Augmentation

Image augmentation is a crucial step in deep learning that increases the diversity of the training data without actually collecting more images. For this model, ImageDataGenerator was configured to perform several augmentation techniques, including horizontal flip, vertical flip, rotation up to 180 degrees, and fill mode adjustment. These augmentations were applied on-the-fly to each batch during training, thereby preventing overfitting and enabling the model to better handle variations in cattle images.

Specifically, the training generator was set up with preprocessing_function = tf.keras.applications.mobilenet_v2.preprocess_input, which scales the pixel values to a range suitable for MobileNetV2. This ensures that the input images match the distribution expected by the MobileNetV2 base model, resulting in optimal feature extraction. The augmentation parameters were carefully chosen to simulate realistic variations that could be encountered in field images, such as changes in orientation and slight movements, improving the robustness of the model.

Model Architecture: MobileNetV2

The core of the CNN model architecture is MobileNetV2, a state-of-the-art, lightweight CNN architecture that is efficient for image classification tasks. MobileNetV2 is particularly known for its use of depthwise separable convolutions and inverted residual blocks with linear bottlenecks, which significantly reduce computational cost without compromising performance. MobileNetV2 was initialized with pre-trained weights from ImageNet, allowing the model to leverage general features learned from a large dataset of natural images.

For this task, the pre-trained layers of MobileNetV2 were frozen, meaning that the weights in these layers were not updated during training. This approach preserves the robust feature extraction capabilities of MobileNetV2 while focusing the training process on learning domain-specific features in the added custom layers. Freezing the base model not only reduces training time but also helps prevent overfitting, especially when the dataset is limited.

On top of the frozen MobileNetV2 base, custom fully connected layers were added to tailor the model for binary classification. The output of MobileNetV2 was connected to a series of dense layers with 'selu' activation functions and dropout layers for regularization. The architecture included two dense layers with 256 units each, followed by two additional dense layers with 128 units. These layers allowed the model to learn higher-level patterns relevant to differentiating between LSD-affected and healthy skin. The final output layer consisted of two neurons with a softmax activation function, producing probabilities for the two classes (Lumpy Skin and Normal Skin).

```
 Layer (type)                   Output Shape         Param #     Connected to
==================================================================================================
 input_3 (InputLayer)           [(None, 224, 224, 3  0           []
                                )]

 Conv1 (Conv2D)                 (None, 112, 112, 32  864         ['input_3[0][0]']
                                )

 bn_Conv1 (BatchNormalization)  (None, 112, 112, 32  128         ['Conv1[0][0]']
                                )

 Conv1_relu (ReLU)              (None, 112, 112, 32  0           ['bn_Conv1[0][0]']
                                )

 expanded_conv_depthwise (Depth  (None, 112, 112, 32  288        ['Conv1_relu[0][0]']
 wiseConv2D)                    )

 expanded_conv_depthwise_BN (Ba  (None, 112, 112, 32  128        ['expanded_conv_depthwise[0][0]']
 tchNormalization)              )

 expanded_conv_depthwise_relu (  (None, 112, 112, 32  0          ['expanded_conv_depthwise_BN[0][0
 ReLU)                          )                                ]']

...
Total params: 2,701,378
Trainable params: 443,394
Non-trainable params: 2,257,984
```

Fig 5.1.4.3 CNN Model Architecture

Model Compilation and Training

The model was compiled with the Adam optimizer and categorical crossentropy loss function. Adam was chosen for its adaptive learning rate, which enables faster convergence compared to traditional stochastic gradient descent. The categorical crossentropy loss is commonly used for multi-class classification problems and was suitable here due to the two output classes.

Training was conducted for 20 epochs with a batch size of 32, as configured in the ImageDataGenerator. The train_images generator fed augmented training images into the model, while the val_images generator provided a validation set to evaluate performance at the end of each epoch. Although an EarlyStopping callback was available to prevent overfitting by stopping training if validation performance did not improve, it was commented out in this run. During each epoch, the model adjusted weights to minimize the categorical crossentropy loss, improving classification accuracy.

Model Evaluation and Visualization of Results

After training, the model's performance was evaluated on both the training and testing sets to assess its generalization ability. The training accuracy achieved was 85.89%, while the testing accuracy was 87.38%, indicating that the model performed well in classifying unseen images. These results demonstrate the model's capability to differentiate between LSD-affected and healthy skin with high accuracy.

The CNN model for LSD diagnosis, built on the MobileNetV2 architecture, demonstrated reliable performance in classifying cattle images based on skin condition. The model achieved an accuracy of 87.38% on the test set, confirming its potential as a diagnostic tool. The combination of MobileNetV2's powerful feature extraction with custom dense layers tailored to the task allowed the model to capture relevant visual patterns effectively.

The use of data augmentation and a carefully designed architecture helped the model generalize well despite a limited dataset. In real-world applications, this CNN model can aid veterinarians and cattle farmers by providing quick and accurate assessments of cattle health based on skin condition. Future improvements may include fine-tuning MobileNetV2 layers or adding more data to further enhance model accuracy and reliability in diverse field conditions.

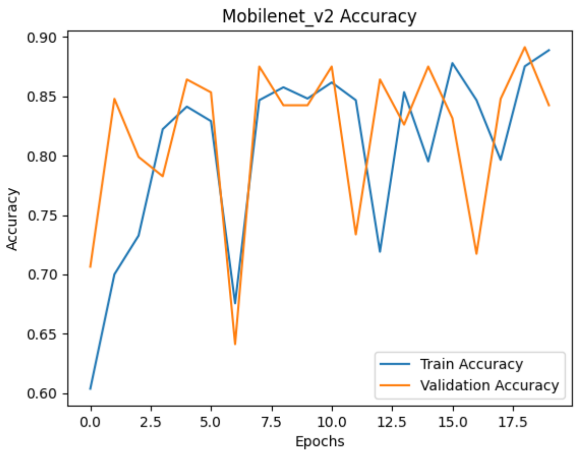

Fig 5.1.4.4 CNN Model Train and Validation Accuracy

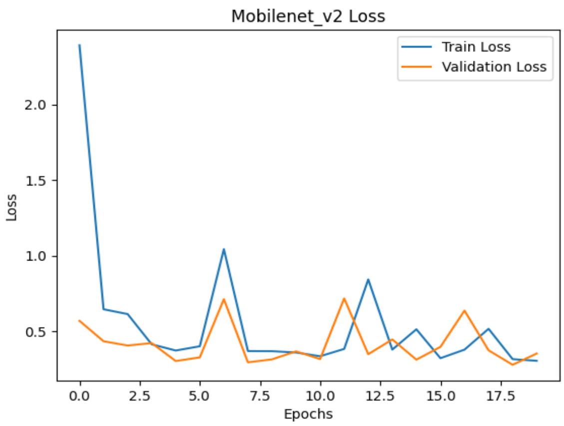

Fig 5.1.4.5 CNN Model Train and Validation Loss

5.1.5 EVALUATION AND PERFORMANCE METRICS

The evaluation and performance metrics for both the ML and CNN models indicate strong accuracy and reliability in predicting Lumpy Skin Disease. The ML models were assessed on metrics including accuracy, precision, recall, and F1-score, with Random Forest emerging as the top performer with an accuracy of 98%, showing balanced sensitivity and specificity. Similarly, the CNN model, built on MobileNetV2, achieved a training accuracy of 85.89% and a testing accuracy of 87.38%, confirming its robustness in classifying LSD-affected and healthy skin images. These results demonstrate that both models are well-suited for deployment, offering reliable diagnostic predictions across structured data and image inputs.

5.2 INPUT AND OUTPUT DESIGN

5.3 DATABASE

The database module is essential for managing and organizing all data within the Lumpy Skin Disease (LSD) diagnostic system. Implemented using MySQL, it securely stores user information, structured clinical data, and image metadata, ensuring efficient retrieval and analysis. Key tables in the database include user data, clinical data, image data, model predictions, and diagnostic reports. This organization allows for easy tracking of each user's inputs, uploaded images, and diagnostic results over time.

Additionally, the database stores feedback from users, enabling continuous improvement of the diagnostic system. With structured data management and security measures like data encryption, the database module provides a reliable backbone for storing, retrieving, and managing data, supporting seamless interactions between the backend and frontend and facilitating accurate and timely diagnostics.

5.4 SYSTEM MODULES

The system for diagnosing Lumpy Skin Disease (LSD) in cattle is composed of several interconnected modules, each serving a distinct function within the overall workflow. These modules work together to ensure efficient data handling, accurate predictions, and clear presentation of results. Below is an overview of the key system modules:

- **Data Collection Module:** This module manages the collection of both structured data (environmental and clinical factors) and image data (cattle skin images) required for LSD diagnosis. Structured data is sourced from reliable platforms like Kaggle, while image data is gathered from repositories such as Mendeley Data. The data collection module ensures that all necessary inputs are available for analysis and prediction.

- **Preprocessing Module:** The preprocessing module is responsible for cleaning and preparing the data to enhance model performance. For structured data, this includes handling missing values, removing outliers, and standardizing features. For image data, the module applies transformations like resizing and normalization, ensuring compatibility with the CNN model. Data augmentation techniques are also applied to the images to improve model generalization.

- **Model Training Module:** This module handles the training of both the ML and CNN models. The ML model uses structured data to learn patterns and make predictions based on clinical and environmental factors, while the CNN model leverages image data to identify visual signs of LSD in cattle skin. Both models undergo hyperparameter tuning and optimization to ensure high accuracy and reliability in their predictions.

- **Prediction and Analysis Module:** Once trained, the models are deployed in the prediction and analysis module. This module takes new inputs—structured data for the ML model and images for the CNN model—and generates predictions on whether a cattle is affected by LSD. The module integrates predictions from both models, providing a comprehensive analysis that combines data-driven and image-based insights.

- **Report Generation Module:** The report generation module compiles predictions from the models into a comprehensive diagnostic report. This report includes the final diagnosis (LSD-positive or negative), key factors influencing the prediction, and recommendations for treatment or management. The report is generated in a user-friendly format to facilitate quick decision-making for veterinarians and cattle farmers.

- **User Interface Module:** The user interface module provides a frontend for data input and results viewing. Users can upload images, enter clinical data, and access diagnostic reports through a web-based interface. Currently, Gradio serves as the frontend platform, allowing for quick deployment and interaction. In the future, the system will be upgraded to a web app with React.js for enhanced functionality and user experience.

- **Database Module:** The database module manages data storage, including user inputs, predictions, and diagnostic reports. Using a structured database like MySQL, this module securely stores and organizes data for efficient retrieval. This storage system allows the application to track historical diagnoses, facilitating longitudinal analysis and potential model improvements based on user feedback.

CHAPTER 6
CONCLUSION AND FUTURE WORKS

6.1 CONCLUSION

The development of a diagnostic system for Lumpy Skin Disease (LSD) in cattle leverages advanced machine learning (ML) and deep learning (CNN) models to provide accurate and timely predictions based on both structured data and images. By combining traditional ML models with a convolutional neural network built on the MobileNetV2 architecture, this system offers a robust, multi-modal approach to diagnosing LSD, addressing a significant challenge in cattle health management. The Random Forest model demonstrated high accuracy on structured clinical and environmental data, while the CNN model effectively distinguished between images of healthy and LSD-affected skin. Together, these models form a comprehensive diagnostic tool that enables veterinarians and farmers to detect the disease early, facilitating quicker intervention and reducing potential losses in livestock.

The system's design incorporates an organized architecture with distinct modules for data collection, preprocessing, model training, prediction, and report generation. Through data augmentation, hyperparameter tuning, and real-time image preprocessing, the models achieved strong performance metrics, validating the effectiveness of this approach. Additionally, the user-friendly interface and secure database implementation allow easy access to diagnostic reports while ensuring data privacy. Overall, this system represents a significant advancement in the use of AI for veterinary diagnostics, offering a scalable solution that can adapt to various field conditions.

6.2 FUTURE WORKS

While the current system offers an effective solution for LSD diagnosis, there are opportunities for further enhancement. Future developments could include fine-tuning the CNN model by unfreezing select layers or integrating newer architectures to improve detection accuracy. Additionally, incorporating real-time data from external sources, such as climate and vector activity, could make predictions more context-aware.

Expanding the system to mobile platforms with IoT-enabled image capture would allow veterinarians and farmers to access diagnostics in real-time, even in remote locations. Moreover, integrating generative AI to generate detailed, easy-to-understand reports could enhance user experience, making the tool more accessible to non-specialists. Finally, the system could be scaled to diagnose additional cattle diseases with similar symptoms, broadening its utility in livestock health management.