International Journal of Advance Research in Science, Management, Engineering and Technology
Volume 1, Issue 5 | November - December 2026 | www.ijarsmet.com
ISSN: 3139-1141

Cattle Care AI

A Deep Learning-Based Framework for Preliminary Cattle Skin Disease Screening

Shruti Indrajeet Patil[#1], Kalashree Parshuram Pakhare[#2], Prof.Vishwas Kenchi[#3]
[#]*Master of Computer Applications, Zeal Institute of Business Administration, Computer Application and Research (ZIBACAR), Pune, Maharashtra, India*
[1]shrutipatilpatil8@gmail.com, [2]kalashreepakhare08@gmail.com, [3]vishwas.kenchi@zealeducation.com

Abstract
Healthy cattle are central to dairy output, animal welfare, and the income of farming families. Lumpy Skin Disease (LSD) is a viral infection that produces skin nodules, fever, and other clinical signs, and noticing the symptoms early can prompt farmers to seek veterinary help sooner. This paper describes Cattle Care AI, a prototype that labels cattle skin photographs as lumpy skin or normal skin and offers preliminary decision support. Its classifier uses transfer learning: a frozen EfficientNetB0 backbone pretrained on ImageNet feeds a small trainable classification head. Input images are resized to 224 by 224 pixels, and augmentation is applied during training only. On an independent test set of 373 images, in which no original photograph appears in more than one subset, the model reached 93.57 percent accuracy and a macro-averaged F1-score of 93.56 percent. The model is embedded in a Streamlit application that offers image upload, prediction display, symptom checking, risk-oriented recommendations, prediction history, image comparison, and report generation. The system is meant as an educational and preliminary screening aid. It cannot confirm a disease, and it does not replace a veterinary examination or laboratory testing.

Keywords—Cattle Care AI; Lumpy Skin Disease; deep learning; EfficientNetB0; transfer learning; image classification; Streamlit; cattle health

I. Introduction

Cattle are a mainstay of agriculture and dairy production, and their health has a direct bearing on milk yield, animal welfare, and household income. Lumpy Skin Disease (LSD) is a viral cattle disease whose signs include skin nodules, fever, and enlarged lymph nodes [1], [2]. Noticing unusual skin changes early can help farmers obtain veterinary advice in good time.

Farmers in some areas have little access to veterinary services, or may be unsure what a visible skin change means. Deep learning image classification could offer an extra way to screen photographs and flag possible problems. Such systems can, however, predict wrongly, particularly when a photograph is unclear or another skin condition looks similar.

Cattle Care AI is a prototype that pairs a deep learning image classifier with a web interface. A user supplies an image; the system prepares it, passes it to the model, shows the predicted class, and adds preliminary guidance through a symptom-checking feature. The project makes the following contributions:

- an image classification pipeline built on EfficientNetB0;
- preprocessing and augmentation of cattle skin images;
- integration of the trained model into a Streamlit application;
- prediction history, symptom checking, recommendation, and report generation features; and
- an evaluation of the classifier on an independent held-out test set, reporting class-wise results and a confusion matrix.

The output of Cattle Care AI is preliminary screening information; it is not a confirmed veterinary diagnosis.

II. Literature Review

A. Lumpy Skin Disease

Lumpy Skin Disease is a contagious viral infection of cattle. Its clinical signs include skin nodules, and it can reduce animal health and productivity. The World Organisation for Animal Health publishes information on LSD, its clinical significance, and its control [1], and clinical studies describe how suspected cases are investigated through examination and laboratory methods [2].

Skin changes that can be seen are often not enough to confirm a diagnosis. A computer vision model can assist with preliminary screening, but a veterinary assessment is still needed.

B. Deep Learning for Disease Classification

Convolutional neural networks (CNNs) are a standard tool for image classification: they learn visual features from images and use them to predict a class. Researchers have applied pretrained models to identify cattle skin conditions and other animal diseases.

Earlier work has tried architectures such as EfficientNet and MobileNetV2 for classifying LSD from images [3]–[5]. The performance reported in those studies depends on dataset quality, how images were collected, class balance, and the evaluation procedure, so their results cannot simply be taken as the performance of Cattle Care AI.

C. Transfer Learning

In transfer learning, a model pretrained on a large dataset serves as the starting point for a new task. Here EfficientNetB0 acts as the feature extractor: its pretrained weights supply initial visual features, and a new classification head is trained on the cattle skin classes.

This avoids training a whole deep network from scratch. Final performance, though, still depends on how good and varied the target dataset is, and on the training process.

D. Research Gap

Screening from images is promising, but several practical issues matter:

- lighting and background can differ from one photograph to another;
- different skin conditions may have similar appearances;
- strong results on one dataset do not guarantee the same results on images from new farms;
- a high prediction confidence is not a clinical diagnosis; and
- a useful application should state its limitations and encourage follow-up with a veterinarian.

Cattle Care AI deals with the software side by combining image classification with a user-facing application. How effective it is must still be tested on independent, properly labelled data.

III. Problem Statement and Objectives

A. Problem Statement

Farmers can need help recognising possible skin problems in their cattle. Careful observation is valuable, but without experience visual signs are hard to interpret. The project aims to offer a simple application that classifies cattle skin images into the classes present in its training dataset and shows the result in an easy-to-read interface.

The system is not meant to replace a veterinary diagnosis; its role is to support preliminary screening and prompt appropriate follow-up.

B. Objectives

The objectives of Cattle Care AI are:

1) to prepare cattle skin images for model training;
2) to build an image classifier using EfficientNetB0 transfer learning;
3) to apply data augmentation so that training images show reasonable variation;
4) to integrate the trained model into a Streamlit application;
5) to display predicted classes together with confidence information;
6) to provide symptom-based preliminary guidance;
7) to support prediction history and report generation; and
8) to evaluate the classifier on an independent held-out test set.

IV. Methodology

A. Research and Development Approach

The development process was implementation-focused and moved through dataset preparation, image preprocessing, model construction, training, application integration, and evaluation.

The development process can be summarised as follows:

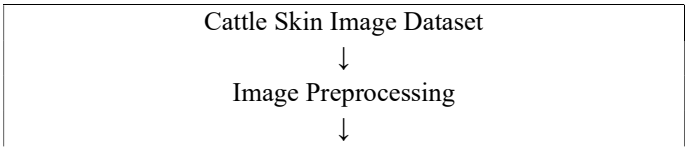


Fig. 1. Main workflow of the Cattle Care AI system.

B. Dataset Description

The project uses the "Cattle diseases datasets" collection from Kaggle [8], which contains three image folders: lumpy (1207 images), healthy (1291 images), and foot-and-mouth (746 images). Only the lumpy and healthy folders were used, so the task is binary. The two target classes are:

1) Lumpy Skin: images from the lumpy folder, which show lumpy skin disease.
2) Normal Skin: images from the healthy folder, which show normal skin.

The class labels are taken from the dataset folder structure [8]. The file names show that many images are augmented variants of the same original photograph: the 2498 images correspond to 2113 distinct original photographs. To prevent variants of one photograph from appearing in different subsets, images were grouped by the original-photograph identifier in their file names. Whole groups were then assigned at random (seed 42) to the training (70%), validation (15%), and test (15%) subsets, separately for each class. A check for exact duplicate files using MD5 hashes found none.

The dataset contains 2498 images: 1207 Lumpy Skin and 1291 Normal Skin images. The numbers of images in each subset are given in Table I.

TABLE I. Dataset Information

Item	Description
Dataset name	Cattle diseases datasets (Kaggle)
Class 1	Lumpy Skin (folder: lumpy)
Class 2	Normal Skin (folder: healthy)
Image format	RGB images
Input image size	224×224 pixels
Training split (70%)	1748 images
Validation split (15%)	377 images
Test split (15%)	373 images
Random seed	42
Total image count	2498
Lumpy Skin images	1207
Normal Skin images	1291
Original photographs	2113

C. Image Preprocessing

Every image is resized to 224 × 224 pixels, the input size the model expects, and is processed with three colour channels (red, green, and blue).

Pixel values are kept in their original 0–255 range because the Keras implementation of EfficientNetB0 contains its own input normalisation. The Streamlit application uses the same

convention, so no additional rescaling is applied before prediction.

D. Data Augmentation

Data augmentation is implemented as layers at the start of the model, so it is applied only during training and is inactive at inference. The transformations are:

- random horizontal flipping;
- random rotation with a factor of 0.08 (approximately ±28.8°);
- random zoom of up to 10%; and
- random contrast variation of up to 10%.

These transformations help the model learn features that do not depend on small shifts in position, scale, and contrast. They are no substitute for a dataset that covers different animals, farms, cameras, and disease stages.

E. EfficientNetB0 Model Architecture

EfficientNetB0 [6], [7] pretrained on ImageNet is used as the feature extraction backbone and its original classification layer is removed. The backbone is kept frozen throughout training; only the classification head is trained and no fine-tuning of the backbone is performed. The classification head contains:

1) Global Average Pooling layer;
2) Dropout layer with rate 0.30;
3) Dense layer with 128 neurons and rectified linear unit (ReLU) activation;
4) Dropout layer with rate 0.20; and
5) Dense output layer with two neurons and Softmax activation, one for each class.

The model can be represented mathematically as:

$$\hat{y} = \text{softmax}(f_{\text{head}}(f_{\text{EfficientNetB0}}(x))) \quad (1)$$

where x represents the input image and $\hat{y}$ represents the predicted class probabilities.

F. Training Configuration

The training pipeline uses the following configuration.

TABLE II. MODEL TRAINING CONFIGURATION

Parameter	Configured Value
Base model	EfficientNetB0
Pretrained weights	ImageNet
Input size	224×224×3
Optimizer	Adam
Initial learning rate	0.001
Loss function	Sparse categorical cross-entropy
Batch size	32
Maximum epochs	25
Early stopping patience	5 epochs
Dropout rates	0.30 and 0.20
LR reduction	Factor 0.2, patience 2
Minimum learning rate	10^{-6}
Backbone	Frozen (no fine-tuning)
Epochs trained	17 (early stopping)
Random seed	42

The training pipeline uses three callbacks:

- EarlyStopping: monitors validation loss and stops training when improvement does not occur for five epochs. It restores the best weights.
- ModelCheckpoint: saves the model with the best validation accuracy during training. The final saved model is the one held at the end of training, which has the best validation-loss weights when early stopping is triggered.
- ReduceLROnPlateau: multiplies the learning rate by 0.2 when validation loss does not improve for two epochs. The configured minimum learning rate is 10^{-6}.

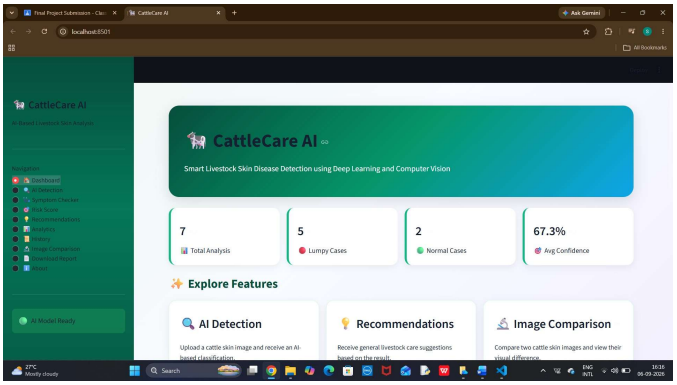


Fig. 2. Cattle Care AI dashboard showing summary cards and feature navigation.

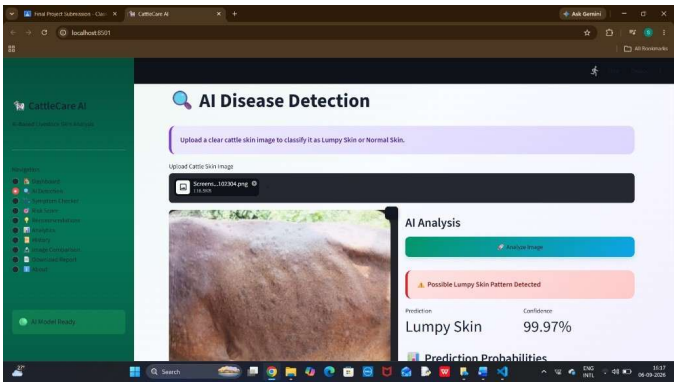


Fig. 3. AI Disease Detection interface showing an uploaded image and model output.

These controls help manage training and keep a useful model checkpoint. They do not by themselves show how accurate the model is on unseen data.

G. Streamlit Application

The model is integrated into a Streamlit application. Its user-facing functions are:

- uploading a cattle skin image;
- displaying the predicted class and confidence information;
- entering observed symptoms;
- displaying preliminary risk-oriented guidance;
- viewing prediction history;
- comparing images or prediction information; and
- generating downloadable reports.

The application links the trained model to a user-friendly interface, and the screenshots below show how it currently

 ISSN: 3139-1141

looks. They illustrate the interface design only and are not evidence of diagnostic accuracy.

Figs. 2–4 show the dashboard, detection, and history screens. The confidence values displayed are outputs of the application and must not be read as clinically validated certainty.

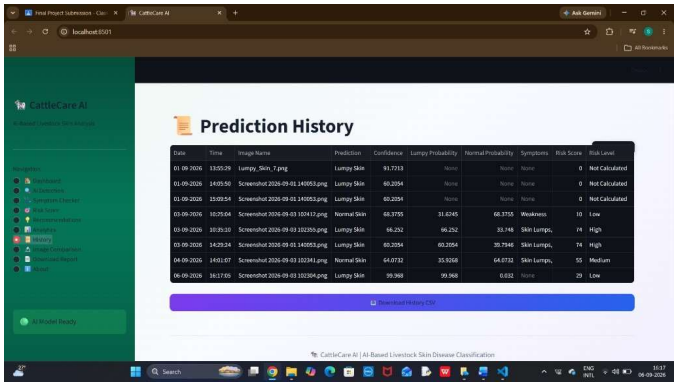


Fig. 4. Prediction History interface showing recorded prediction entries.

H. System Requirements

The project is written in Python. Streamlit provides the application interface, and image-processing and data-handling libraries support the application.

The experiments used Python 3.12, TensorFlow 2.18.0 with Keras 3 for the neural network, scikit-learn 1.9.1 for the evaluation metrics, and Streamlit 1.66.1 for the interface.

V. Findings and Evaluation

A. Implementation Findings

The project combines an EfficientNetB0-based classification pipeline with a Streamlit application. Implementation covered preparing input images, training the classification head, saving the model, and linking predictions to the user interface.

These points describe the implemented design. Predictive performance is reported in the next subsection.

B. Quantitative Evaluation

The model was evaluated on three sets: training, validation, and an independent test set. The test set has 373 images (180 Lumpy Skin and 193 Normal Skin), set aside before training and never used for model selection. Accuracy and macro-averaged precision, recall, and F1-score were calculated with scikit-learn; predictions and true labels were collected from the same unshuffled datasets so that they remain aligned. Early stopping ended training after 17 epochs.

Table III gives the results. On the independent test set, the model reached an accuracy of 93.57% and a macro-averaged F1-score of 93.56%. Training accuracy (98.05%) is roughly 4.5 percentage points above test accuracy, which points to some overfitting to the training images.

TABLE III. Performance on the Training, Validation, and Independent Test Sets (Macro-Averaged)

Metric	Training	Validation	Test
Number of images	1748	377	373
Accuracy (%)	98.05	94.69	93.57
Precision (%)	98.04	94.70	93.58
Recall (%)	98.09	94.74	93.63
F1-score (%)	98.05	94.69	93.56

Among the 180 Lumpy Skin test images, 172 were classified correctly and 8 were missed (false negatives), giving a sensitivity of 95.56%. Among the 193 Normal Skin test images, 177 were classified correctly and 16 were wrongly labelled lumpy (false positives), giving a specificity of 91.71%. The area under the ROC curve (ROC-AUC) on the test set was 0.9877. In a screening setting, the 8 missed Lumpy Skin images are the more serious error, since an affected animal might be judged healthy.

Accuracy is the share of all predictions that are correct. Precision is the share of positive predictions that are right, and recall is the share of actual positive cases that are found. The F1-score is the harmonic mean of precision and recall.

In disease screening, false negatives and false positives deserve attention alongside overall accuracy, so a confusion matrix and class-wise results are reported for the test set.

C. Application Behaviour

Figs. 2–4 show the application in use. It displays the uploaded image, returns the predicted class and its confidence, logs each prediction in the history screen, and warns that veterinary confirmation is needed. No user study was conducted, so the usability of the application has not been evaluated.

D. Comparison With Published Research

Earlier studies have examined deep learning models for LSD classification, including pretrained CNNs and other architectures [3]–[5], and they provide background for this project. A direct numerical comparison, however, would need matching datasets, class definitions, data splits, and evaluation methods.

For that reason, this paper makes no claim that Cattle Care AI outperforms models from published studies. A fair comparison needs a consistent experimental setup and is left for future work.

VI. Discussion

Cattle Care AI packages image classification and a web application into one prototype. The interface is meant to let users without deep learning experience run the prediction process easily.

EfficientNetB0 transfer learning gives a structured way to extract image features. Augmentation adds variety to the training images, callbacks govern the training run, and the Streamlit application lets users interact with the saved model in a practical way.

How useful the system is, however, depends on its real performance on unseen images. Image quality, illumination, camera distance, background patterns, coat patterns, and the look of other skin conditions can all affect predictions, and the model might learn irrelevant background features rather than the lesion itself.

The application should be clear about uncertainty. Treatment decisions should not rest on a predicted class or confidence value alone, and suspected disease calls for professional veterinary assessment and proper testing.

VII. Limitations

The main limitations of this work are:

1) The study uses two classes and 2498 images, which come from 2113 original photographs. The dataset is a user-uploaded Kaggle collection whose image sources, collection conditions, and label verification are not documented.
2) The model was evaluated on a single dataset; its performance on images from new farms, breeds, cameras, and disease stages has not been measured.
3) Images were grouped by the original-photograph identifier in their file names, but different photographs of the same animal or scene may still appear in different subsets, which can overstate performance.
4) The backbone was kept frozen, and no fine-tuning or comparison with other architectures was carried out.
5) The classifier may confuse LSD with visually similar conditions.
6) The symptom-based score has not been established as a clinically validated diagnostic score.
7) The system does not provide laboratory confirmation, prescribe treatment, or replace a veterinarian.

VIII. Future Work

Future development may include:

- collecting more images from different farms, breeds, and disease stages;
- obtaining labels verified by veterinary professionals;
- evaluating the model on images from independent farms and sources;
- checking whether predictions focus on meaningful skin regions using an explainability method such as Grad-CAM;
- fine-tuning the backbone and comparing EfficientNetB0 with other suitable architectures;
- validating the symptom-based guidance with veterinary professionals; and
- testing application speed and reliability on the intended deployment device.

A later version could also look at using extra information safely, such as an animal's history and observed symptoms. Such extensions would need suitable data, validation, and careful handling of user information.

IX. Conclusion

Cattle Care AI is a prototype for preliminary screening of cattle skin images. It combines an EfficientNetB0 transfer learning model with a frozen backbone, image augmentation, validation-based training controls, and a Streamlit application that displays predictions and supports reports. On an independent test set, the model achieved an accuracy of 93.57% and a macro-averaged F1-score of 93.56%.

The project shows how deep learning can be built into an accessible application for animal-health screening. Results from a single public dataset, however, do not establish clinical reliability; validation on images from independent farms, with veterinary-confirmed labels, is needed before practical use.

Until the system has been properly evaluated, it should be used only for educational and preliminary decision-support purposes. It does not replace a veterinary examination or laboratory diagnosis.

Acknowledgment

The authors acknowledge the academic guidance and learning resources that supported the development of this project.

References

[1] World Organisation for Animal Health, "Lumpy skin disease." [Online]. Available: https://www.woah.org/en/disease/lumpy-skin-disease/

[2] M. Mathewos, F. Dulo, Z. Tanga, and M. Sombo, "Clinicopathological and molecular studies on cattle naturally infected with lumpy skin diseases in selected districts of Wolaita Zone, Southern Ethiopia," *BMC Veterinary Research*, vol. 18, 2022, doi: 10.1186/s12917-022-03403-4.

[3] S. M. Saqib *et al.*, "Lumpy skin disease diagnosis in cattle: A deep learning approach optimized with RMSProp and MobileNetV2," *PLOS ONE*, vol. 19, 2024, doi: 10.1371/journal.pone.0302862.

[4] C. Senthilkumar, S. C., G. Vadivu, and S. Neethirajan, "Early detection of lumpy skin disease in cattle using deep learning—A comparative analysis of pretrained models," *Veterinary Sciences*, 2024, doi: 10.3390/vetsci11100510.

[5] M. Pal *et al.*, "Implementation of a deep learning system for detection and classification of lumpy skin disease in cattle: Enhancing precision and efficiency in veterinary diagnostics," *Veterinary Medicine and Science*, vol. 11, 2025, doi: 10.1002/vms3.70664.

[6] TensorFlow, "EfficientNetB0." [Online]. Available: https://www.tensorflow.org/api_docs/python/tf/keras/applications/EfficientNetB0

[7] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proc. 36th Int. Conf. Machine Learning (ICML), 2019, pp. 6105–6114.

[8] devang03mgr, "Cattle diseases datasets," Kaggle. [Online]. Available: https://www.kaggle.com/datasets/devang03mgr/cattle-diseases-datasets (accessed Oct. 26, 2026).