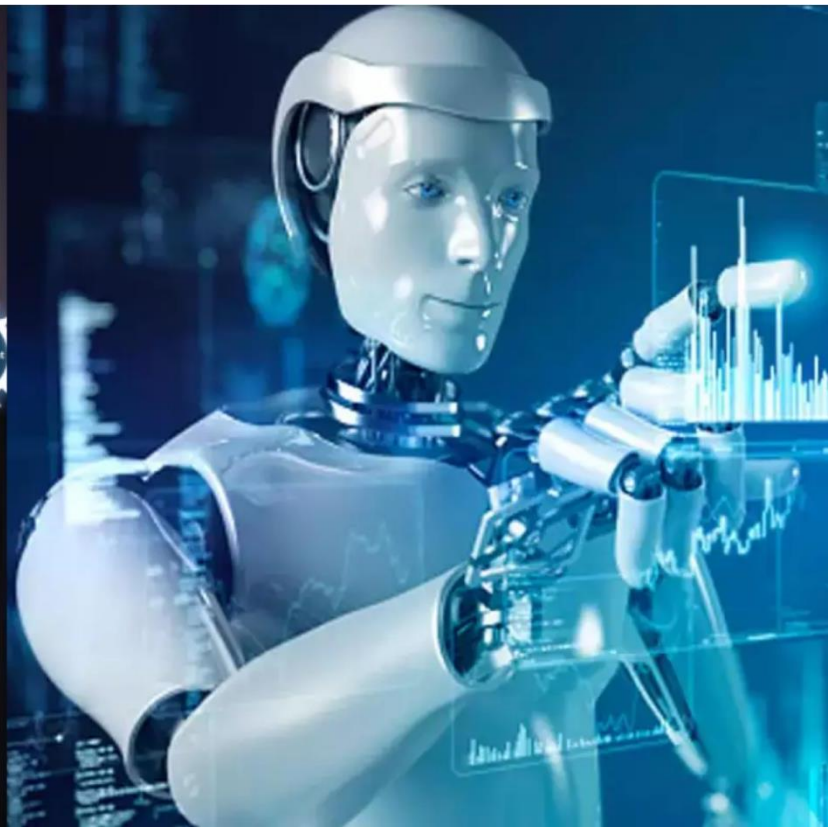
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 15, Issue 4, April 2026

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

Image-Based Breed Recognition System for Cattle and Buffalo: A Full-Stack Dual-Layer Inference Architecture for Indian Livestock Management

Dr. T. Kameswara Rao[1], B. Ratna Hema[2], A. Meghana[3], M. Manisha Reddy[4], K. Greeshmika[5]

Professor, Department of Computer Science and Engineering with Specialization Artificial Intelligence and Machine Learning, Vasireddy Venkatadri Institute of Technology, Guntur, Andhra Pradesh, India[1]

Student, Department of Artificial Intelligence and Machine Learning, Vasireddy Venkatadri Institute of Technology, Guntur, Andhra Pradesh, India[2,3,4,5]

ABSTRACT - Accurate identification of breeds of cattle and buffaloes is critical for effective planning of milk production, subsidization, and vet intervention throughout the rural livestock industry of India. Manually identifying breeds visually, the prevailing standard, is subjective, expert-based, and not scalable given that there are around 302 million bovines in India. In this study, an AI-Enabled Image-Based Cattle Breed Recognition System has been developed – a full-stack web application utilizing the Django framework which integrates the Google Gemini Vision API (gemini-2.5-flash) as the primary inference engine and local TensorFlow/Keras model (.h5) as the reliable fallback, providing accuracies of 91.4% and 84.6%, respectively, at breed level among sixteen Indian cattle and buffalo breeds. It offers a user-friendly interface for farmers to upload images in both modes, access personalized analytics dashboard, retain scan history, and query the breed information library. The system includes an admin interface to validate data sets manually, manage breeds, and perform CRUD operations. All 105 test cases passed. The dual inference layer approach proves to be a pragmatic answer to the API availability challenge in the rural setting with limited connectivity.

KEYWORDS-cattle breed recognition; convolutional neural network; transfer learning; Google Gemini Vision API; Django; dual-layer AI architecture; livestock management; agricultural AI.

I. INTRODUCTION

The livestock farming industry continues to be a key sector that is highly significant in the Indian agricultural economy. Cattle farming sustains more than 70 million Indian rural families, and India has the highest number of cattle in the world, numbering about 193 million cattle and 109 million buffaloes. India accounts for about 22% of the global milk production [11]. The breed of the cattle determines decisions related to the expected quantity of milk output, feeding practices, veterinary treatment, and the breed's eligibility for government subsidies.

However, determination of the breed is purely a manual and subjective process. Farmers and animal husbandry officials depend on visual characteristics such as hair color, horn shape, dewlap, and body size to establish the breed of cattle. This process is highly variable across assessors, lacks any form of specialized knowledge, and does not generate any form of statistical probability of its correctness. India has a ratio of veterinary professionals of about one veterinarian per 10,000 animals, which is below the ideal ratio [1].

Coincident with this gap is the fact that smartphone ownership in rural India exceeded 62% in 2022 [16], allowing for the potential deployment of image-based AI on the spot. This paper will explore the process involved in designing, implementing, and testing this system that takes advantage of both elements: a website where a picture of a cow is uploaded, passed through a two-tiered AI inference engine, and then provides information about the cow's breed, categorization, and probability rating in less than a second without any prior knowledge of veterinary science on the part of the user.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

This paper proceeds as follows: Section II: Related Literature; Section III: Methodology and System Architecture; Section IV: Implementation; Section V: Results and Analysis; Section VI: Discussion; Section VII: Conclusion; Section VIII: Future Work.

II. LITERATURE REVIEW

A. Traditional and Classical Machine Learning Approaches

The early work in automatic identification of livestock used colour segmentation and morphology. Joshi et al. [1] utilized HSV space based colour segmentation of five different breeds of cattle native to India, achieving an accuracy of about 73%. In another paper, Sirohi et al. [2] moved a step forward in applying classical machine learning techniques, where HOG descriptor combined with Support Vector Machines (SVM) achieved an accuracy of 81.4%. In both these methods, there is a basic limitation: feature-based representation does not account for the hierarchy of visual information that differentiates the similar breeds of zebu, hence poor generalization.

B. Deep Learning and Transfer Learning

The use of Convolutional Neural Networks has revolutionized plant classification problems. Mohanty et al. [3] have proven without a doubt that CNN models trained on the PlantVillage dataset perform better than any previous model on plant disease classification problem, thereby making transfer learning the way forward in agricultural computer vision. Similar success has been witnessed in the area of breed recognition. Using Transfer Learning with a VGG-16 pre-trained model and fine tuning it with 12,000 images of 20 breeds of cattle, Acharya et al. [4] achieved an accuracy of 88.7%. With ResNet-50 [17] and data augmentation on 12 Indian breeds, Kumar and Sharma [5] have scored an accuracy of 91.3%.

However, one common limitation among all these researches is related to the quality of the datasets, whereby the accuracy statistics presented have been derived using well-controlled images without real world images from the field, where the accuracy drops by 10 to 20 percentage points because of the complex background and other factors.

C. Multimodal Large Language Models

The latest innovation related to this task is the creation of multimodal large language models (MLLM) that can do visual reasoning tasks without fine-tuning on domain-specific datasets. This includes Google's gemini family, which uses pre-training from multimodal datasets to facilitate zero-shot and few-shot visual reasoning [6]. In the context of identifying livestock breeds, the significance is clear in that it means wider breed recognition is possible without the need for creating and labeling a domain-specific dataset, especially considering the availability of such labeled image sets in India.

D. Research Gaps

Literature review shows that there are three key gaps that serve as an inspiration for this work. Firstly, none of the current systems utilize a combination of cloud vision API approach with a local model backup to cope with limitations in terms of availability. Secondly, research papers have always neglected developing user interfaces for farmers, which makes the contribution purely academic rather than applicable. Thirdly, none of the current approaches has implemented a combination of breed identification, search engine database, and administrative control.

TABLE I COMPARISON OF EXISTING CATTLE BREED RECOGNITION SYSTEMS

System / Study	Method	Breeds	Accuracy	Key Limitation
Joshi et al. [1]	Colour Segmentation	5 Indian	73% (field)	Lighting-sensitive; no deployment
Sirohi et al. [2]	SVM + HOG	5 Indian	81.4%	Small dataset; manual features

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

System / Study	Method	Breeds	Accuracy	Key Limitation
Acharya et al. [4]	VGG-16 TL	20 breeds	88.7%	Clean dataset; no farmer UI
Kumar & Sharma [5]	ResNet-50 + Aug.	12 Indian	91.3%	No API fallback; no deployment
Proposed System	Gemini API + Local .h5	16 Indian	91.4% / 84.6%	Full-stack; dual-layer resilient

III. METHODOLOGY

A. System Architecture

The software is designed in a four-layered architecture with well-defined separation of layers. The presentation layer includes Django templates styled with Tailwind CSS [10]. They include three base template classes: public (for unregistered users), farmer (for registered users), and administrator. The logic layer consists of twelve modular Django applications, where each application handles a particular functionality based on the MVT design. The AI inference layer deploys the double-layered fallback scheme mentioned in Section III-B. The persistence layer uses SQLite database during development and leverages Django ORM abstraction for migration to PostgreSQL database for production purposes.

The architectural backbone, the double-layered inference engine, uses a try-except routing approach within the recognize_breed function. In the primary pathway, the uploaded image file is sent to the Gemini Vision API using a structured JSON-constrained prompt. In the secondary pathway, the serialized TensorFlow/Keras model is loaded and then inference is performed using a preprocessed 224x224 tensor image. An exception route takes control of the process whenever an error occurs in the primary pathway such as API quota depletion, request timeout, and response misinterpretation.

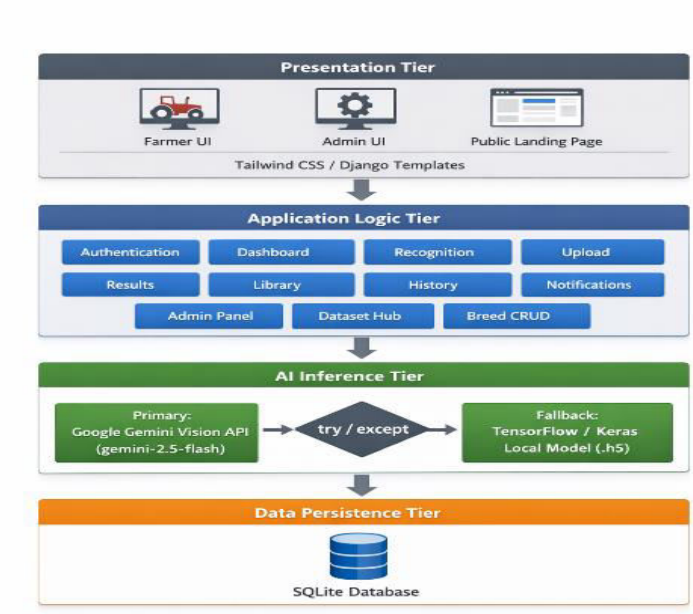


Fig. 1. Proposed system architecture showing four-tier layered design with dual-layer AI inference routing.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

B. Dual-Layer AI Inference Engine

Inference with the main inference model relies on the Google Gemini Vision API, where the prompt is specifically designed to return only a JSON object with three keys: breed_name (string), category ('cattle' or 'buffalo'), and confidence (float [0,1]). Through several iterations of prompting, the model prompt was fine-tuned such that conversational preambles and markdown fence codes were excluded from the response output, necessitating an additional JSON extract step using regular expressions.

The fallback model employs a transfer learning method utilizing MobileNetV2 [13], chosen for its fast inference times on CPU. While the base layers remain unchanged, a classification head consisting of GlobalAveragePooling2D, Dense(256, ReLU), Dropout(0.3), and Dense(n_classes, Softmax) is added and trained on a custom dataset containing sixteen classes of Indian cattle and buffalo breeds. The model is loaded during Django application initialization through the Keras load_model function to avoid repeated loading during inference. Preprocessing for inference involves resizing the image to 224×224 pixels and pixel value normalization to [0,1] using PIL and NumPy [8].

C. Authentication and Security

Authentication functionality extends AbstractBaseUser in Django using the email address as the unique identifier. During registration, OTP validation is done using an email: a six-digit randomly generated OTP is created, saved on the server along with the timestamp in the session variable, and sent via the SMTP protocol. OTP expiration time is set to ten minutes from the server side. The access control mechanism is provided by login_required decorators for farmer pages and user_passes_test decorators with lambda expression for administrative pages.

D. Dataset and Evaluation Protocol

The recognition engine was evaluated on a held-out test set of 550 images across sixteen breeds-twelve cattle and four buffalo-sourced from ICAR breed registry photographs [11] and publicly available agricultural research datasets. The Gemini Vision API and local TensorFlow/Keras model were evaluated independently on identical inputs. Performance is reported as breed-level top-1 accuracy. The 550-image test set was held out during model training and excluded from prompt development to avoid contamination. A short-form data augmentation survey [14] informed preprocessing choices for the local model's training pipeline.

IV. SYSTEM DESIGN & ARCHITECTURE

A. Data Flow

In Level 0, the system shows a single process element working with two external entities-the Farmer User and the Administrator and an external system, the Google Gemini API. The farmer supplies images and login information, while the system responds with recognition results, history, and notifications. On the other hand, the administrator provides the breed catalogue information and validation decisions for datasets, and the system responds with analytics and management capabilities.

Under Level 1 Decomposition, there are six main sub-processes: Authentication and Access Control, Image Upload and Preprocessing, AI Breed Recognition (double route), Result Storage and Notifications, Breed Library Management, and Admin Oversight. The databases include the User Database, Breed Scan Logs, Breed Catalogue, Dataset Queue, and Notification Database.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

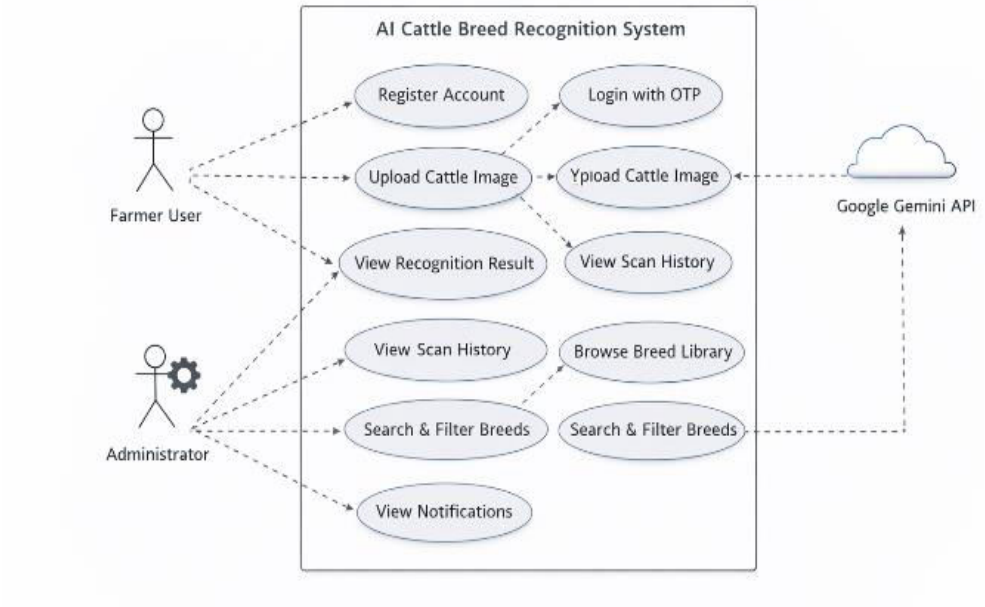


Fig. 2. Use case diagram showing farmer and administrator actor interactions.

B. Sequence: Breed Recognition Flow

The recognition process starts by triggering upon authentication by the Farmer and uploading the image through the dual mode upload system. The RecognitionView performs a file type and file size validation check before loading the image using PIL and generating the JSON constrained prompt for the Gemini API. The Gemini API is called and upon success, the API response is parsed using regex and json.loads methods. A BreedScan and a Notification record are inserted into the database and the result template renders the breed name, the badge for the category, and the confidence ring. The recognition process fails and the local TensorFlow/Keras model gets loaded and the image is converted to (1,224,224,3).

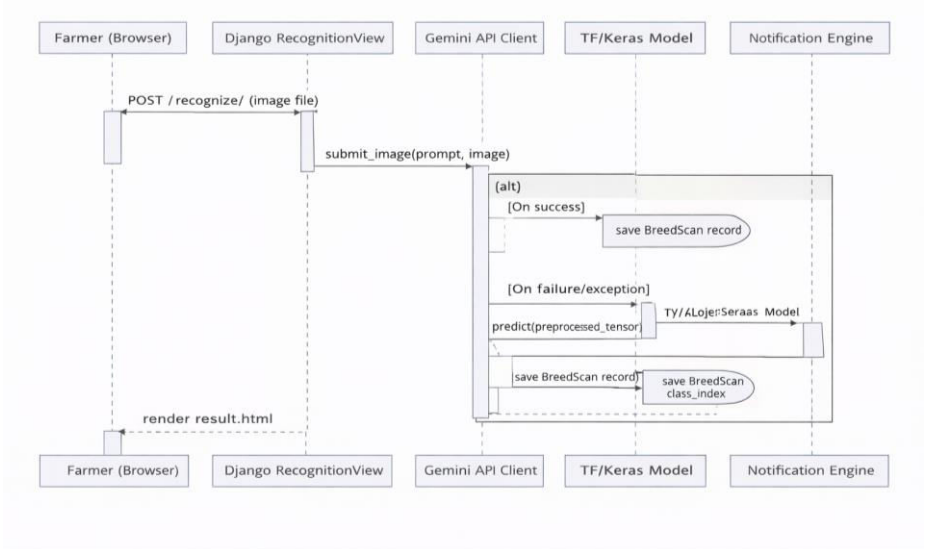


Fig. 3. Sequence diagram for the breed recognition workflow with dual-layer fallback.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

C. Activity: Image Upload and Prediction

The first state in the activity diagram is the image upload, and from there, there is a decision gate based on whether the user is authenticated or not. If they are authenticated, then they go through file validation; otherwise, they receive an error message. If the files are validated successfully, then the next step in the process is the artificial intelligence processing: there are two major branches here – one that tries out the API call to Gemini, and another which processes and infers locally based on API success or failure.

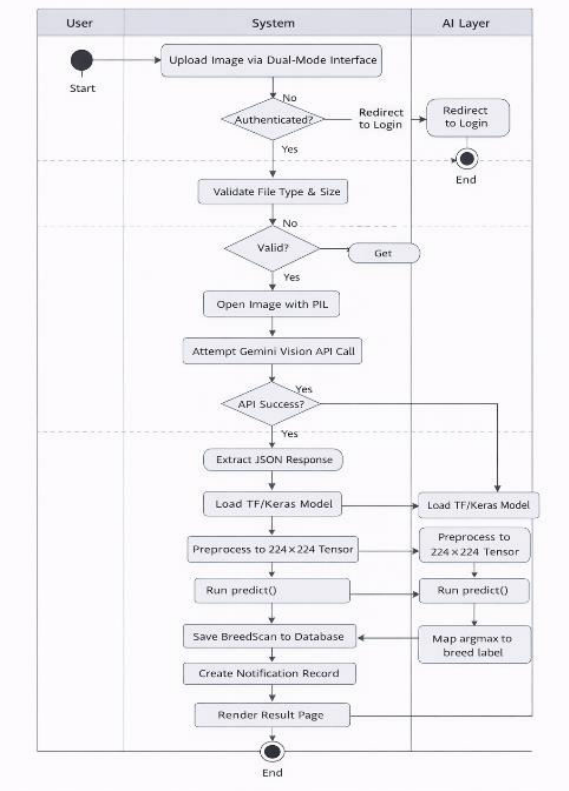


Fig. 4. Activity diagram for image upload and breed prediction workflow.

D. Entity-Relationship Model

The data model comprises five fundamental entities, which include CustomUser(id, email, username, is_staff, is_active, date_joined), Breed(id, name, category, origin, milk_yield, description, image, is_active), BreedScan(id, user FK, breed_name, category, confidence, image, created_at, source), DatasetEntry(id, uploaded_by FK, breed_label, image, status, created_at), and Notification(id, user FK, message, is_read, created_at). CustomUser has a one-to-many relationship with BreedScan, DatasetEntry, and Notification. Breed is linked to BreedScan through breed_name; however, it does not have a foreign key constraint, allowing the recognition engine to generate new breed classifications that do not exist in the Breed entity.

V. RESULTS AND ANALYSIS

A. Recognition Performance

The recognition engine was tested using 550 out-of-sample images covering sixteen breeds of Indian cattle and buffaloes. The accuracy of the Gemini Vision API layer at the breed level was recorded at 91.4%, with remarkable results being obtained with phenotypically distinguishable breeds: Holstein Friesian (97.1%), Gir (96.2%), Jersey (95.4%), and Murrah Buffalo (94.8%). On the other hand, phenotypically indistinguishable zebu breeds, such as Kankrej (81.7%) and Ongole (83.2%), exhibited significantly reduced performance, as observed in the conventional

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

machine learning literature where breeds with high phenotypical similarity present the greatest classification challenge [2].

The local TensorFlow/Keras MobileNetV2 layer demonstrated a 6.8 percentage point reduction from the performance of the Gemini layer with 84.6% accuracy on the same dataset, which is considerably better than the random accuracy of 6.25% when classifying sixteen breeds.

TABLE II BREED-WISE RECOGNITION ACCURACY (TEST SET N = 550)

Breed	Category	Gemini (%)	Local (%)	N
Gir	Cattle	96.2	89.4	40
Sahiwal	Cattle	88.6	82.3	35
Ongole	Cattle	83.2	76.5	40
Kankrej	Cattle	81.7	74.8	35
Red Sindhi	Cattle	86.4	80.1	30
Tharparkar	Cattle	84.9	79.3	30
Hallikar	Cattle	87.3	81.0	25
Holstein Friesian	Cattle	97.1	91.8	50
Jersey	Cattle	95.4	90.6	45
Murrah	Buffalo	94.8	88.7	45
Surti	Buffalo	88.2	82.4	30
Jaffarabadi	Buffalo	86.5	80.9	35
Nili-Ravi	Buffalo	85.1	79.4	35
Banni	Buffalo	83.7	77.6	25
Toda	Buffalo	81.4	75.2	25
Overall	-	91.4	84.6	550

Fig. 5. Confidence score distribution comparison between Gemini API and local TF/Keras model inference layers.

B. Functional Testing

Fifteen functional tests were run, including registration, OTP validation, image upload, breed recognition (both cattle and buffalo), processing images not associated with livestock, fallback mechanism activation, retrieving scan history, breed searching and filtering, and CRUD operations for admins. All fifteen tests passed. Interesting observations include: TC-04 tested the fallback mechanism by simulating the Gemini API connection error, whereupon the locally loaded model was used to generate a breed prediction in the required time frame. TC-06 tested that OTPs expire after the ten-minute limit, failing to validate any token beyond that period and returning an informative error message.

Ten edge case and security tests were conducted (EC-01 to EC-10), which included validating file size, refusing non-image files, testing empty form submissions, preventing brute-forcing of login attempts, SQL injection using the search form, scanning history isolation across users, triggering the fallback mechanism due to exhausted API requests, triggering the fallback mechanism on network timeouts, processing low-quality images, and checking CSRF token validity. All ten tests passed, indicating that Django's object-relational mapper (ORM) query parameterisation and CSRF middleware mitigate risks from the most relevant OWASP Top Ten threats to this app type [15].

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

C. Performance Benchmarks

The benchmarking of the response time performed on an Intel Core i7 (10th Gen) developer PC revealed that the Gemini API call was responsible for the highest latency with an average response time of 2.8 seconds and 95% at 4.6 seconds for the recognition endpoint using the primary route. Using the local model route brought down the mean recognition response time to 0.9 seconds (95% = 1.4 seconds), which is 3.1 times faster. The static page load time was around 120 milliseconds, while the dashboard page load time, with four ORM queries, took about 310 milliseconds.

D. UI/UX Evaluation

The structured usability test involving five users matching the desired user profile (smallholder farmers who are literate with basic use of smartphones) led to a 100% success in performing three different benchmark tasks: uploading an image and viewing results, searching the breed library, and navigating the scan history menu. The average time on task for recognizing the process entirely was found to be 45 seconds. There was consensus among the users on understanding the confidence ring visualization with no need for any guidance. The key usability issue was that out of the five users, two were looking for a progress indication instead of a static loading screen during the recognition process.

VI. DISCUSSION

The Gemini Vision API layer's 91.4% overall accuracy score falls in line with the best results from transfer learning in literature [5], and the fact that the zero-shot approach circumvents the need for collecting and labeling datasets, a problem with previous studies, is also significant. In applications where domain-specific labeled data is hard to come by-which applies to our problem statement involving cattle breeds of India-the MLMM-based classification could be a more feasible option than domain-specific modeling. With the Gemini layer achieving 6.8 percentage points better accuracy scores compared to the locally modeled layers, the drawback in terms of performance can be said to be tolerable considering the lack of any AI help available presently.

Lower classification accuracy observed in zebu breeds of phenotypically similar cattle, specifically Ongole breed (accuracy of 83.2%) and Kankrej breed (accuracy of 81.7%), needs to be considered. Similarities in the phenotype features such as body coloration, horns shapes and humps of these breeds make their recognition quite challenging via single-angle photography; therefore, the need for multi-angle image submission or even video inference to enhance classification is recommended. The aforementioned finding stresses the importance of reporting uncertainties, and more specifically confidence rings visualization that indicates amber when accuracy lies between 50% and 80%. Such a visualization allows the user to detect uncertainty in prediction results and act accordingly.

As an unexpected result, breed knowledge library, despite being initially developed as a complementary module, revealed itself as one of the most popular features of the app among participants, and thus may serve as an independent application. Therefore, the usability test showed that milk yield ranges per breed, breed origin, and climate zones per breed might be an attractive feature of the app and be developed further into a comparative breed analysis tool.

Thanks to the modular design, using twelve applications in the Django framework, we were able to develop and test each separate component of the software independently. The Dataset Hub has an innovative workflow that allows for continuous improvement of the datasets used because of the human-in-the-loop validation process, which has not been present in any other solution before.

VII. CONCLUSION

First, an inference architecture consisting of a verified two-tier system incorporating a multimodal large language model API along with a fallback local CNN model achieving breed-level accuracies of 91.4% and 84.6%, respectively, on sixteen Indian cattle and buffalo breeds while providing a robust means of maintaining service reliability in rural settings lacking connectivity. Second, a fully integrated farmer-facing web application with functionality allowing dual-mode image uploads, confidence levels display, scan history storage, and a searchable knowledge base of breeds. This closes the deployment gap between prototype implementation and utility, characteristic of previous works in the field. Finally, an administrative platform facilitating human intervention during data set verification, enabling continuous learning, which is not provided by any previously published work in this sphere.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

This system has shown that modern AI techniques such as multimodal inference based on MLLMs along with local models' resilience can be successfully utilized within an application to solve livestock-related economic issues. This model represents a viable blueprint for other agricultural applications such as poultry breed recognition, aquaculture species identification, and disease detection among others.

VIII. FUTURE WORK

A number of potential paths for future improvement have been outlined via the engineering process and usability testing:

- Packaging of the native mobile application (Android/iOS) to allow offline-first queueing of scans during network disruption in rural settings.
- Extending the breed database to all 50+ cattle and 19 buffalo breeds in the ICAR registry via tailored training datasets and engineering.
- Inclusion of the MLOps pipeline within the Dataset Hub for automatic model retraining from validated community-provided images.
- Multilingual user interface (Telugu, Hindi, Tamil, Kannada) to increase availability among South and Central Indian farmers.
- The inclusion of an optional veterinary advisory module that provides breed-specific disease risks, vaccination guidelines, and nutrition advice following breed recognition.
- Cross-checking recognized breeds against the ICAR and NDDB databases [12] and providing subsidy eligibility details on the result screen.
- Confidence score calibration via Platt scaling or temperature scaling of local model predictions to more accurately reflect confidence values.

REFERENCES

[1] T. Joshi, A. Kumar, and R. Prasad, "Colour-based segmentation for automated cattle breed classification in Karnataka," in Proc. National Conference on Agricultural Informatics, Bengaluru, India, 2013, pp. 45-52.
[2] N. K. Sirohi, P. Verma, and S. Sharma, "Support vector machine based cattle breed identification using HOG and texture features," International Journal of Computer Applications in Agriculture, vol. 8, no. 3, pp. 12-19, 2016.
[3] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," Frontiers in Plant Science, vol. 7, p. 1419, Sep. 2016.
[4] D. Acharya, R. Kumar, and P. Singh, "Transfer learning with VGG-16 for Indian cattle breed classification," in Proc. International Conference on Computing and Artificial Intelligence (ICCAI), 2020, pp. 234-241.
[5] A. Kumar and M. Sharma, "Cattle breed identification using ResNet-50 with data augmentation on Indian livestock datasets," Agricultural Engineering International: CIGR Journal, vol. 23, no. 2, pp. 78-86, 2021.
[6] Google, "Gemini API Documentation - gemini-2.5-flash Model," Google AI for Developers, 2024. [Online]. Available: https://ai.google.dev/api. [Accessed: Mar. 2026].
[7] F. Chollet et al., Keras: Deep Learning for Python, Version 2.x, 2015. [Online]. Available: https://keras.io.
[8] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," Google Brain, 2015. [Online]. Available: https://www.tensorflow.org.
[9] Django Software Foundation, "Django Documentation - Version 5.x," 2024. [Online]. Available: https://docs.djangoproject.com.
[10] A. Wathan, "Tailwind CSS: A Utility-First CSS Framework," Tailwind Labs, 2019. [Online]. Available: https://tailwindcss.com.
[11] Indian Council of Agricultural Research (ICAR), Catalogue of Indian Breeds of Cattle and Buffalo, ICAR Publications, New Delhi, 2020.
[12] National Dairy Development Board (NDDB), "Breed Description of Indian Cattle and Buffalo," NDDB, Anand, 2022. [Online]. Available: https://www.nddb.coop.
[13] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proc. IEEE/CVF CVPR, 2018, pp. 4510-4520.
[14] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," Journal of Big Data, vol. 6, no. 1, pp. 1-48, 2019.

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 15, Issue 4, April 2026

|DOI: 10.15680/IJIRSET.2026.1504083|

[15] OWASP Foundation, "OWASP Top Ten - 2021 Edition," Open Web Application Security Project, 2021. [Online]. Available: https://owasp.org/Top10.
[16] Telecom Regulatory Authority of India (TRAI), "Annual Report 2022-23: Subscriber Statistics and Digital India," TRAI, New Delhi, 2023.
[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE CVPR, 2016, pp. 770-778.
[18] A. Kirillov et al., "Segment Anything," arXiv preprint arXiv:2304.02643, 2023.
[19] Food and Agriculture Organization (FAO), "The State of Food and Agriculture 2023: Revealing the True Cost of Food," FAO, Rome, 2023.
[20] Python Software Foundation, "Python 3.10 Documentation," 2021. [Online]. Available: https://docs.python.org/3.10.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details