AI-Driven Livestock Identification and Insurance Management System

Abstract

Cattle identification is pivotal for many reasons. Animal health management, traceability, bread classification, and verification of insurance claims are largely depended on the accurate identification of the animals. Conventionally, animals have been identified by various means such as ear tags, tattoos, rumen implants, and hot brands. Being non-scientific approaches, these controls can be easily circumvented. The emerging technologies of biometric identification are extensively applied for Human recognition via thumb impression, face features, or eye retina patterns. The application of biometric recognition technology has now moved towards animals. Cattle identification with the help of muzzle patterns has shown tremendous results. For precise identification, nature has awarded a unique Muzzle pattern that can be utilized as a primary biometric feature. Muzzle pattern image scanning for biometric identification has now been extensively applied for identification. Animal recognition via Muzzle pattern image for different applications has been proliferating gradually. One of those applications includes the identification of fake insurance claims under livestock insurance. Fraudulent animal owners tend to lodge fake claims against livestock insurance with proxy animals. In this paper, we proposed the solution to avoid and/or discard fraudulent claims of livestock insurance by intelligently identifying the proxy animals. Data collection of animal muzzle patterns remained challenging. Key aspects of the proposed system include: (1) the Animal face will be detected through visual using YOLO v7 object detector. (2) After face detection, the same procedures will apply to detect muzzle point (3) the muzzle pattern is extracted and then stored in the database. The System has a mean average precision of 100% for the face and 99.43% for the nose/muzzle point of the animal. Once the animal is registered in the database, the identification process is initiated by extracting unique nose pattern features with ORB and/or SIFT. Then it is matched using the pattern matchers like BFMatcher and/or FLANNMatcher for animal identification. The proposed model is more efficient and accurate as compared to concurrent approaches. The results extracted from this research study show 100% accurate identification.

Keywords: Machine Learning, Transfer Learning, Deep Learning, Artificial Intelligence

1. Introduction

Nature provided human beings with countless blessings and animals are one of those. They are beneficial in many ways as a source of nurtured food like meat, dairy products, and clothing such as woolen and leather [1].

To protect the Cattle effectively from multiple hazards, their accurate identification is of utmost importance. Cattle identification can broadly be categorized into two types namely contact and non-contact methods [2]. Under the contact method, various techniques have been used manually to recognize animals such as hot iron branding, freeze branding, ear tags, rumen implants, tattoos, etc. [3]. Though conventional methods served the purpose to some extent but with a major pitfall of easy circumvention [4]. Contact methods inherent major drawback of Animal stress due to the painful procedure involved to mark identity. However, these are invasive for the cattle and non-cost-efficient as compared to other more effective processes [5]. Non-contact methods include AI-based technologies using biometric identification and other machine learning models.

Cattle identification is equally important for Livestock Risk coverage. Insurers are highly concerned about the exact recognition of insurable animals; not only to provide adequate cover but also to establish a firm basis for subsequent verification. Cattle identification using conventional approaches remain a significant issue for livestock insurers and veterinarian for registration and health monitoring activities.

Manual processing of cattle identification is considered highly vulnerable to fake insurance claims that can't be detected accurately [4].

During the last couple of decades, with the evolvement of AI technologies such as machine learning / deep learning, human recognition using human biometrics features of the face, iris, fingerprint, etc. It has become quite easy to recognize human beings with greater accuracy [6]. The recognition using biometrics is not limited to humans anymore and now animal recognition has also become a vital segment of this emerging technology [7].

The cattle identification system could be used to fulfill a variety claims, animal traceability, registration, monitoring, and

Animal Registration through a Biometric system, based on scientific techniques is widely used for Cattle identification and traceability [8]. A highly transparent process of cattle registration would mitigate the risk of manipulation, fraudulent verification, and cattle swapping. The Cattle Registration process comprises a standard framework-based system [9] for verification of bogus insurance claims of registered cattle. Not only for Insurance Claims verification, but it's also equally important for the implementation of animal safety policies [10].

Biometric identification of cattle provides fundamental information for numerous applications. One of the prominent applications is to manage fake livestock Insurance claims [11]. Farm owners who have insured their cattle, unfortunately, tend to make false claims because a non-technical layman who did not stay around the cattle so much, cannot easily recognize the cattle. Only those who stay for a longer period and are appropriately familiar with the cattle can identify which one had been insured. Hence, these fraudulent farm owners successfully deceive Insurers and obtain claim money for the cattle that were not even registered. To overcome this issue of bogus Insurance claims lodged by farm owners, an Automated Intelligent Cattle Identification and Insurance Claim Management System is proposed to overcome the most critical issue faced by the insurers which are fake claims. Just like human fingerprints, cattle muzzle print comprises two unique features recognized as ridge bifurcation and ridge termination which can be used to track accurate Identity by matching relevant information with preserved datasets.

The input parameters of the proposed system comprise three layers namely Data Acquisition, Pre-processing, and Data Preparation. The Data Acquisition Layer represents the source from which the data is firstly pooled in the system. In this segment, two activities will have been performed namely collection of raw images of animals and then storing cluster datasets into the database management system. We also use a video dataset to enhance accuracy. The Number of frames in the video sequence impacts the identification results.

Pre-processing is an essential layer applied to extract image features and subsequent matching procedures for object recognition [12]. The prime objectives to be achieved in the pre-processing segment are to minimize though not eliminate the noises and other particles muzzle images. Next to data collection, data will be purified by applying certain processes. Major activities In Preprocessing segment include to Remove Blurry and noisy images, Cropping and Resizing of images. Data Preparation is the final stage of Data input for the biometric system of Cattle identification. Purified data received from preprocessing layer will finally tag in the Data preparation layer. Activities performed include labeling/ notation of images and then splitting data into train and test data.

The proposed system will intelligently detect the face and muzzle point of the cattle, and identify/recognize the cattle using muzzle patterns in real-time.

The system will not only be helpful for the detection of false insurance claims but it can also be utilized on farms for monitoring the cattle, for their health, safety, and management. The traditional approaches to recognizing cattle except for the naked eye are tagging the animal using ear tags, and implanting microchips in the animal [13]. While microchips like NFC, RFID, etc. are better than ear tags, they can be expensive, and hurtful to the cattle as they will be needed to be implanted inside the cattle and there will be a need for an expert who can safely execute the said procedure, which can be expensive. On the other hand, using ear tags is not very efficient as these tags can be forged or switched from one animal to other pretty easily.

The proposed system offers wide range of potential applications, including the accurate and efficient tracking of individual cattle for breeding or medical purposes, as well as monitoring their behavior and movement within feedlots, and identifying lost or stolen animals. This novel approach has the potential to revolutionize cattle management and tracking, resulting in increased productivity and efficiency within the agricultural sector. By utilizing advanced computer vision techniques, the proposed system provides a rapid and precise method for identifying specific cattle based on their unique biometric characteristics, offering significant advantages over traditional identification methods.

Section 2 demonstrates the literature review. Section 3 explains the Research Methodology to intelligently identify cattle in real-time. It includes description of all the adopted tools and technologies for achievement of our research objectives. Section 4 elucidates the performance of proposed model and experimental results. Section 5 concludes the research work and recommend some new avenues for future researches.

2. Literature Review

During the last couple of decades, numerous research studies have been conducted on biometric features based on Animal identity [8]. The emergence of Precision Livestock Farming (PLF) played an important role in the fourth Industrial Revolution [14]. Owners of big cattle farms are induced to adopt new identification techniques because traditional methods are more expensive, cumbersome to implement, and sometimes inefficient to manage a larger number of animals. Moreover, Artificial Intelligence (AI) has made the task much easier to identify any cattle with greater accuracy, minimal effort, and shorter time [15]. Since ancient times, various techniques have been employed for accurate identifications of cattle. Conventional systems of cattle recognition include Hot or cold branding, ear tagging, ear tattoos, and ear notches [16]. Though these are comparatively cheaper the process is very paining for cattle which often causes severe health issues for the animals [16].

Alternatively, the Biometric system uses visual features to identify an individual animal. Biometric features may include iris patterns, muzzle images, and eye retina. These methods though introduce computerized systems but they also involve some critical challenges, such as accuracy based on extracted features and time required for process completion. Anders Herlin et al. [17] highlighted the usage of digital technologies such as GPS collars, E-tagging, and Chips used for RFID and its implications for the management and monitoring of cattle farms.

The recent work in cattle face recognition has brought forth multiple techniques to recognize cattle faces. In earlier days of research on cattle biometrics features, Scale Invariant Feature Transform (SIFT) technique had been used to extract muzzle patterns features from image print [18]. The implanted RFID chips are also used for tracking and identification of cattle. However, it is challenging for large herds to implement RFID effectively [4]. Researchers used texture fusion techniques for the extraction of muzzle features. Worapan Kusakunniran et al. [19] proposed a fusion of Transition Local Binary Pattern with Gabor feature. Convolution Neural Network was proposed for greater accuracy of image recognition. The CNN technology is applied on basis of system training instead of a hardcoded algorithm; resultantly a large number of images can be identified [20]. Kumar et al. [21] proposed cattle face recognition approach using LBP and SURF feature extractors with Gaussian Pyramid [22] levels 1 through 4. This technique might not work properly with images in different lighting conditions. Also, they are using full face images of cattle which is not very efficient in recognizing cattle as mostly the cattle have similar facial features. Zin et al. [23] proposed a method of cow detection using ear tags, they used you only look once (YOLO) [24] object detection to detect cow head in real-time, and from there, they recognize ear tags using image segmentation techniques. The problem with this technique is the ear tags that can be altered and forged. Hongyu Wang et al. [25] proposed a parametric transfer approach with VGGFace dataset and VGG-16 deep convolutional neural network. Pre-trained VGG-16 network is then fed with their cattle face dataset. Their dataset contains images of cattle faces, which again is not a very efficient way to recognize cattle, especially buffalos, as in low-lit environments, their facial features cannot be differentiated. Santosh Kumar et al. [26] proposed a system to recognize cattle in real-time. The proposed system recognizes cattle using the muzzle point or nose pattern of the cattle. It involves: (1) the cattle is captured through a video camera in real-time, (2) the frames are then cropped into nose/muzzle point images (3) which are then preprocessed further to remove any noise or blurry images and converted into grayscale images, (4) then their features are extracted using appearance-based feature extractor algorithms, (5) which are then stored into the database. In their testing phase, they take the nose image as a query image and then extract features using FLLP extractor. At this point, the features are matched with the features in the database, where a threshold value is returned, which is compared with a predefined threshold value to determine the decision of cattle recognition. The problem with this method is that nose images are cropped from surveillance video frames. Santosh Kumar et al. [27] presented a comprehensive review of cattle identification along with problems/drawbacks involve in various methods, they emphasized on development of a framework for cattle identification via muzzle point. Table 1 has the comparison of proposed solution with related work by other researchers.

Table 1: Related Work Comparison with Proposed Solution

Paper/Author	**Title**	**Method Used**	**Accuracy**	**Limitation**	**Our Solution**
Kumar, Santosh & Singh, Sanjay. (2015).	Face Recognition for Cattle	LBP and SURF Feature Extraction along with Gaussian Pyramid Smoothing. The Algorithms that are used are: Principle Component Analysis Linear Discriminant Analysis Independent Component Analysis ICA_LibSVM	92.75%	The Data they have used is the photos of cattle faces. The cattle have very similar faces and hence should not be used for identification.	We are recognizing cattle by using cattle biometrics that is their nose pattern, as each animal has a unique nose pattern.

Kumar, Singh Sanjay. (2016).	Automatic identification of cattle using muzzle point pattern: a hybrid feature extraction and classification paradigm	They extracted muzzle point texture features using: Haralick texture features techniques morphology based features, shape based features Histogram of Oriented Gradient (HOG) Wavelet Colour features Tamuras Laws Texture Energy (LTE) Local Binary Pattern texture Fuzzy-Local Binary Pattern The Algorithm they used for classification: KNN Fuzzy KNN Radial Basis Function Decision Tree Gaussian Mixture Model Probabilistic Neural Network MLP Classifier Naïve Bayes Classifier	96.74% with Fuzzy-KNN	They have used muzzle point recognition and the system requires capturing muzzle point images of cattle which is not an easy thing to do as animals tend to run away.	Considering the issue of animals running away, our system lets the laymen point the camera towards the animal and it will automatically capture images, and not just muzzle point, but the animal, its face and muzzle, which can be used for even better recognition/identificatio n of the animal.
Kusakunnira n et al., (2020)	Biometric for Cattle Identification using Muzzle Patterns	contrast limited adaptive histogram equalization (CLAHE) technique to enhance muzzle image extracted from cattle face. Then BoHoG and LBP histogram are used for recognition	95.13%	They have used watershed algorithm and estimated the position of the muzzle point at the center of the image which will not be true in real-time and cannot always keep the animal have its muzzle point in the center.	This issue is also dealt with in our system as with our system all it is needed to keep the camera pointed at the animal and it will automatically capture images, and not just muzzle point, but the animal, its face and muzzle, which can be used for even better recognition/identificatio n of the animal.
Wang, Qin, Hou and Gong, (2020)	Cattle Face Recognition Method Based on Parameter Transfer and Deep Learning	VGGNet Deep Convolutional Neural Network. Pre-trained and fine-tuned ImageNet	70%	They are using face images and no animal biometrics are involved.	We are recognizing cattle by using cattle biometrics that is their nose pattern, as each animal has a unique nose pattern.
Thi Thi Zin et al.(2020)	Cow Identification System Using Ear Tag Recognition	You Only Look Once (YOLO) base cow Head detection. Images Segmentation for Ear Tag recognition.	96% Cow Head Detection 84% Cow recognition	They are using ear tags to recognize the cows, which is not an efficient approach as the	Our system has 100% mAP for cattle head/face detection, and our approach is based on animal biometrics.

				tags can be forged and switched.	
Yongliang Qiao et al.(2021)	Automated Individual Cattle Identification Using Video Data: A Unified Deep Learning Architecture Approach	CNN BiLSTM. Inception-V3 CNN for visual feature extraction	93.3%	Accuracy based on video frame length may not improve or even decrease after a certain frame length	We use VOLOv7 model for object detection which is the most efficient in speed and accuracy.
Ali Shojaeipour	Automated Muzzle Detection and Biometric Identification via Few-Shot Deep Transfer Learning of Mixed Breed Cattle.	YOLO-ResNet-50	99.11%	They have used a smaller dataset of 563 images. Further studies are required to verify the accuracy.	In the proposed model we use 8,020 images which is reasonable sample size for generic prediction.

3. Research Methodology

The proposed system to intelligently identify cattle in real-time or in still images or videos requires an image dataset of cattle to be trained so that it can detect the face/head and nose of cattle. The process of collecting image datasets for training and testing consists of the following steps, which include:

- Data Acquisition
- Data Preprocessing
- Data Augmentation
- Data Preparation

Data Acquisition

Raw images of cattle were taken using different smartphones with different camera capabilities. The environments in which the images were taken were different too, to add diversity to image data. These include indoor, outdoor, sunny, and cloudy weather conditions. The cattle include domestic buffalos, cows, calves, and bulls. Taking pictures of these animals was the most difficult part of data acquisition as it required a lot of hard work and running after the animals. This struggle resulted in a lot of useless images, a few of which can be seen in Fig. 1. The difficulty in taking the pictures of cattle was that the animals would get scared the moment a person goes near to grab their picture, as it is required for the dataset that the images of cattle should be clear and have their head/face in it. Another problem that occurred was that, because some of the cattle were eating/grazing, their nose/muzzle point was covered in grass straws, hence taking such images would be of no use. Even though it was the priority to collect clear and concise images of cattle, a lot of useless images were still captured along with the required images. Images of animals including, cows, buffalos, bulls, and calves along with images other than these were collected, which included humans, dogs, sheep, horses, and donkeys, etc.

Figure 1: Noise in Data

A dataset of 300 animals from Shojaeipour, Ali et al. [28] was also added in the dataset which totaled in 5875 images, which were not much, as the dataset still required some cleansing.

Data Preprocessing

Once the data was collected, the images were then filtered out. Some irrelevant visuals which are considered noise in the data can be seen in Fig. 1, had to be deleted, some images needed cropping, and some images needed to be resized. This step was necessary to have images that are required to properly train the system.

Data Augmentation

Data augmentation is a technique where, when required, more data is generated from existing data if the existing data is not enough. That is the case with our dataset and thus in order to increase the data set a bit, data augmentation was used. The augmentation steps include: horizontal flip, -10 degrees to 10 degrees rotation, crop, grayscale, hue, brightness, exposure, noise and saturation. Doing this substantially increased the amount of the dataset.

Data Preparation

Now when the data is preprocessed, and augmented the next step was to prepare it for training and testing. Our purpose is to detect the face/head, nose of cattle, dirty nose, and faces other than cows/buffalos. So, for detection in images, image data has to be annotated to be trained for object detection. A total of 9400 images were annotated using the open source image Labeling Tool [29]. Once annotated, it had to be split into training ~94%, validation ~5%, and testing 1 %, Figure 2.

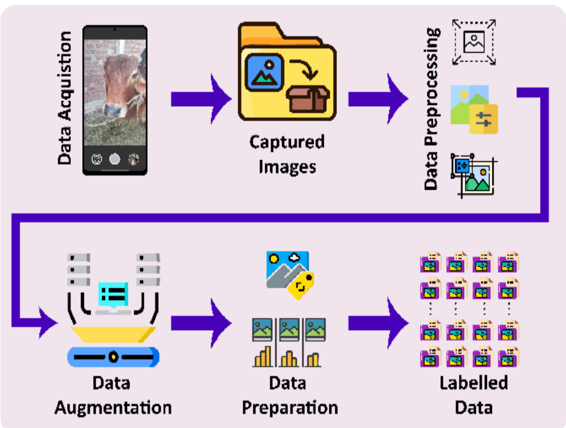

Figure 2. Data Preparation

We have used five classes namely: Face, Nose, Nose-Dirty, and Not-cow, Table 2.

- Face:

The class represents cattle face. It is necessary to detect cattle face to locate muzzle point, and to make sure a cow or buffalo is present in the scene.

- Nose:

Detection of Nose/Muzzle Point is the ultimate objective to extract muzzle pattern for biometric identification of the cattle, thus Nose class.

- Nose-Dirty:

In case the muzzle point of the animal has something on it which can cause the muzzle pattern or the lower lip of the animal to not be visible, in that case, it would be useless to extract muzzle pattern and the identification process can be flawed, thus this class is necessary to avoid any dirty muzzle pattern to be captured, and the system will be able to alert about the dirty muzzle of the animal.

- Not-cow:

Some object detection models are likely to detect other animal faces as cow/buffalo faces, and in some case, they even detect human faces so Not-Cow class is used as shown in Table 2 to make sure the model understands the difference, and to make sure the subject which is a Cow/Buffalo is in the scene and its face is being captured to get its nose pattern.

Table 2: Data Distribution and Class Labels

Class	Train		Validation		Test	
	Image	Label	Image	Label	Image	Label
All	8820	14463	480	786	100	167
Face	8820	4945	480	269	100	56
Nose	8820	3342	480	181	100	39
Nose-Dirty	8820	2878	480	178	100	38
Not-Cow	8820	3298	480	158	100	34

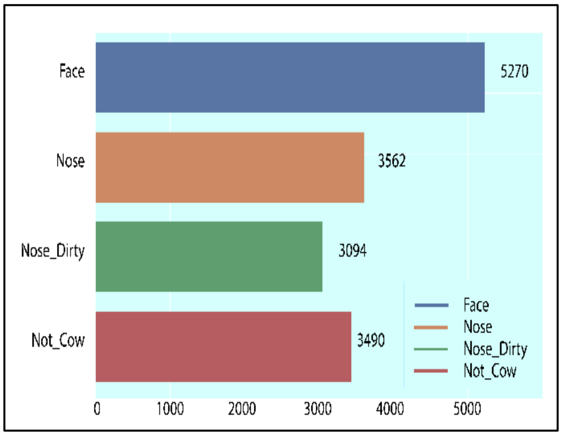

Figure 3. Class Labels Analysis

Class label balance can be seen in the following analysis of training data, Figure 3. Total annotated dataset comprises on 15,416 class labels representing all the four classes.

Training

Now that we have our dataset annotated and prepared as shown in Fig. 4, the dataset is now ready to be trained.

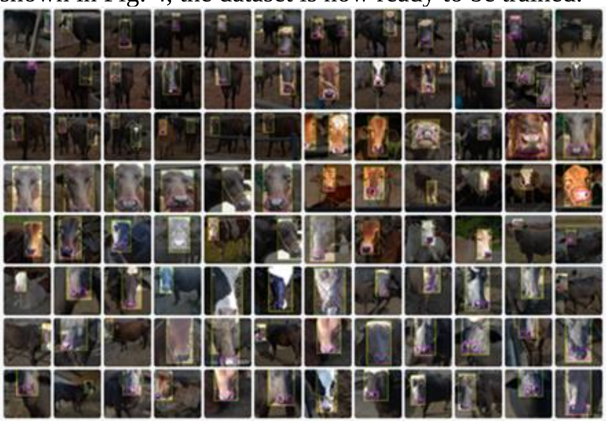

Figure 4. A Sample from Annotated Data

Detecto

We used a pre-trained Fast R-CNN ResNet-50 FPN model-based python package Detecto [30]. It is a python package based on PyTorch and can be used for the training of object detection. However, it did not prove to be efficient for our problem even though above 80% accuracy was achieved Figure 5 shows training loss. But when the model was tested it would detect the face with above 90% accuracy.

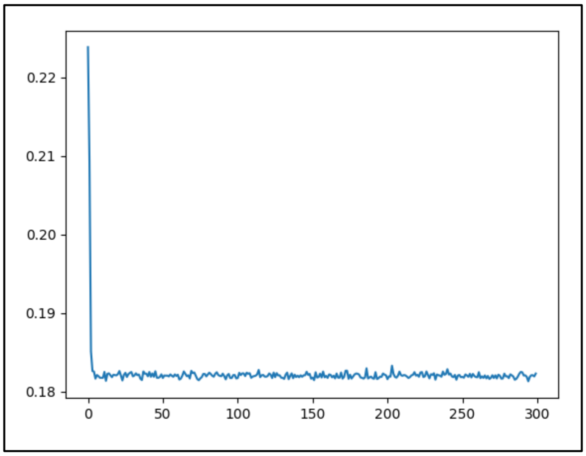

Figure 5 Detecto Loss Curve

It seemed to mostly leave the nose part out, which would be a problem as our main goal is to detect the nose once the face was detected, Figure 6. Detecto model does not properly predict the bounding box. The bounding box predicted by the model is way off the required subject which in our case is the cattle head/face, its nose, either dirty or not, and faces other than cattle, including humans.

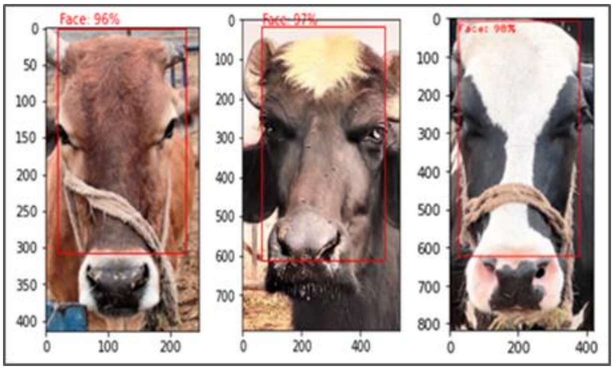

Figure 6: Cattle Face Detection using Detecto

Yolov4

We have also trained on yolov4 for object detection, which shows considerably good mAP and no overfitting, following Figure 7 shows the training loss and mAP gain. Even though yolov4 showed promising results with mAP of 100% yet it was not considered final model for detection as its inference time was high.

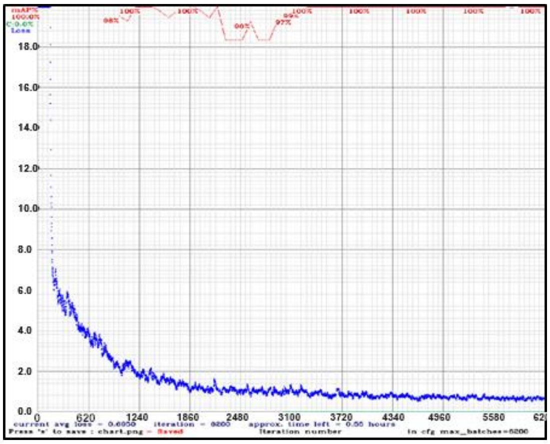

Figure 7 Yolov4 mAP – Loss

Yolov7 - Network Architecture

YOLOv7 presents the latest technology to support the multiple object tracking (MOT) framework [31]. Yolov7 performs extremely well when it comes to object detection. It detects objects more accurately and swiftly than the previous versions. In this architecture enhanced cardinality of new features has been achieved by using collective convolution while the gradient-oriented routing does not change as shown in Fig 8.

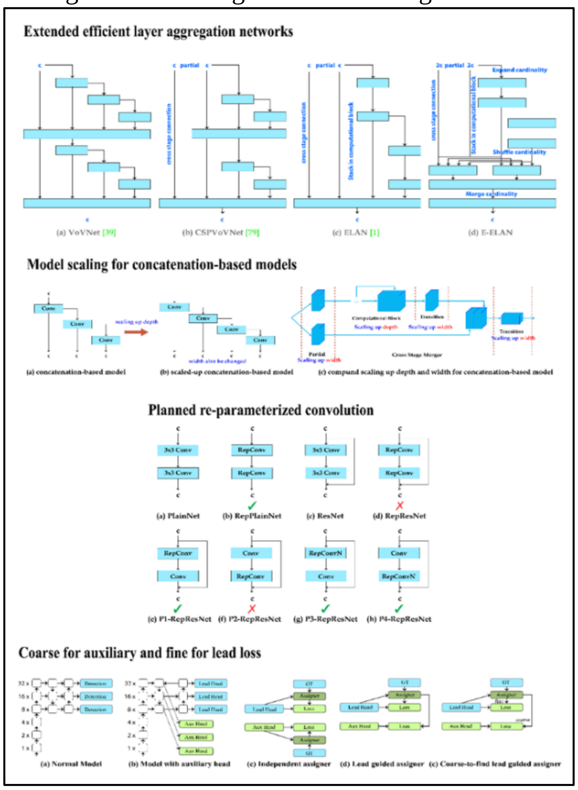

Figure 8:Yolov7 Model Architecture

The process can increase learned features with the help of feature mapping, shuffling, and merging in a cardinality manner. Consequently, improve calculation and parameter usage**.** YOLOv7 is the most appropriate object detector for the aggregate scaling approach. It can be applied to interlinked model architecture to compute modifications in the output width of the computational block. It is recommended to scale only depth in computational blocks. This approach can maintain the initial model design and structure properties. Re-parameterization techniques involve averaging a set of model weights to create a model that is more robust to the general patterns that it is trying to model. In research, there has been a recent focus on module-level re-parameterization where the piece of the network has its re-parameterization strategies. The YOLOv7 authors use gradient flow propagation paths to see which modules in the network should use re-parameterization strategies and which should not.

A state-of-the-art Yolov7 object detection model is the latest introduction to YOLO family. It has a few variants. Some of which we used for training our data are:

- Yolov7-tiny (training from scratch)
- Yolov7x or Yolox (training using transfer learning)
- Yolov7 (training from scratch)

a. Yolov7-tiny

We performed training using transfer learning of pre-trained yolov7 tiny model and it can be seen in the following Fig 9, the overall mAP achieved by yolov7-tiny on our dataset is 0.9946 or 99.46%. The loss curve is seen in the Fig 10.

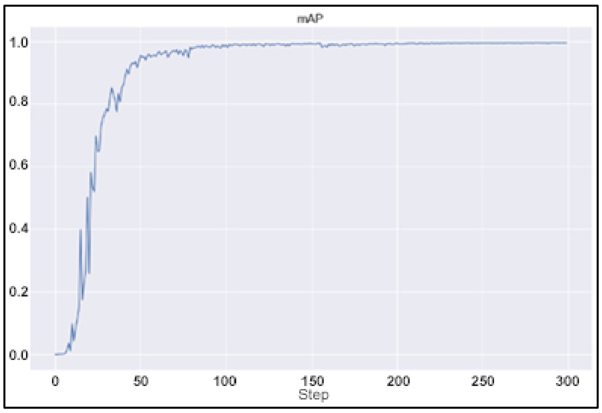

Figure 9 YOLOv7-tiny mean Average Precision

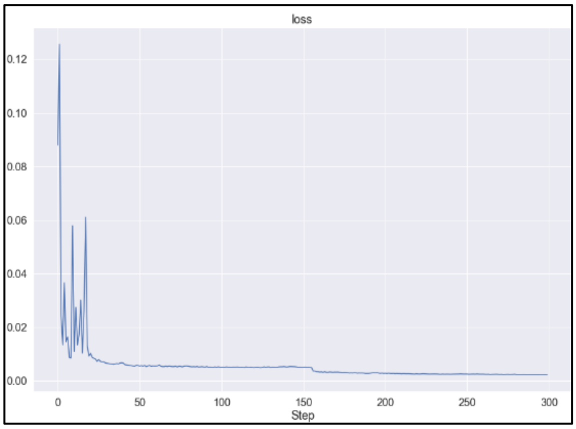

Figure 10 YOLOv7-tiny Loss Curve

There is no overfitting of the model as can be seen in the figures above and test results are present in Fig 11.

Figure 11 Yolov7-tiny Predicting Bounding Boxes on Test Set

b. Yolov7x

We also trained yolov7x model on our dataset and the training was done from scratch, following figures: Fig 12 & Fig 13 show the achieved overall mAP and loss curve respectively. Yolov7x mAP is around 78% and the loss curve shows over fitting.

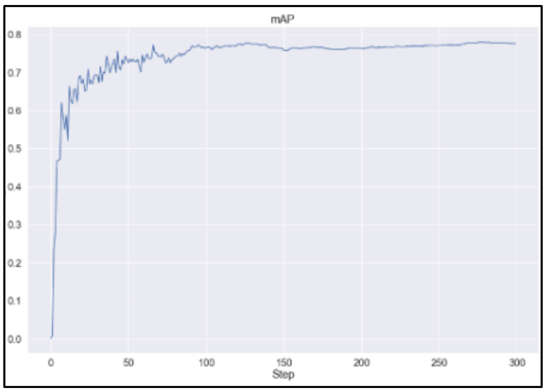

Figure 12 Yolov7-x mean Average Precision

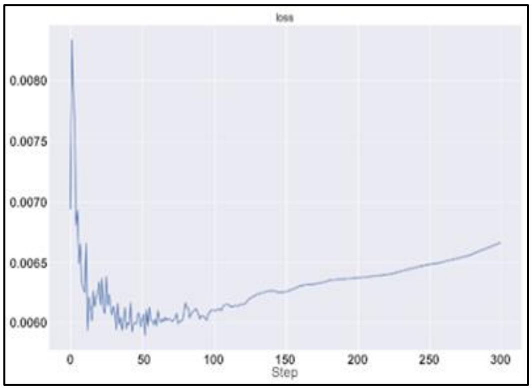

Figure 13 Yolov7-x Loss Curve

c. Yolov7

The Fig 14 show how yolov7 model trained on our data and this yolo variant was able achieved 83% mean average precision after being trained from scratch.

The YOLOv7 model is trained using a deep neural network that is specifically designed for object detection tasks. It comprises on three main segments: the backbone network, the neck network, and the detection head. The backbone network extracts high-level features from the input image, which are important in identifying objects. The neck network refines the extracted features and integrates data from different scales and resolutions, which improves the model's accuracy. Ultimately, the detection head produces the final predictions, generating bounding box predictions and class probabilities for identified objects using the improved features. These components are optimized using cutting-edge methods and convolutional layers to ensure optimal performance in object detection.

Overfitting is visible in Fig 15, loss curve of yolov7 shows slight overfitting of the model.

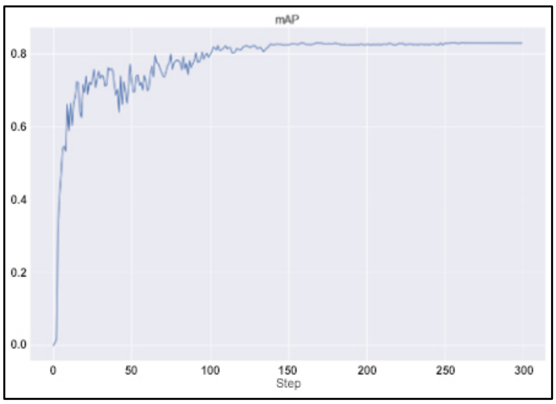

Figure 14 Yolov7 mean Average Precision

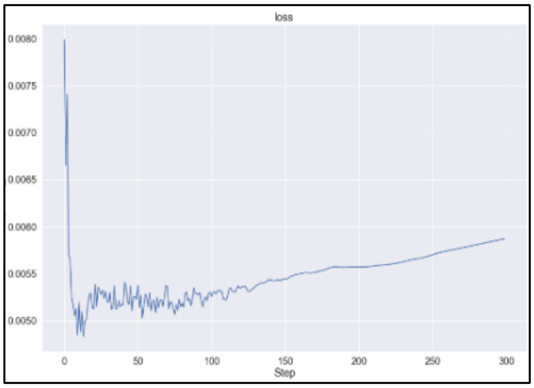

Figure 15 Yolov7 Loss Curve

Once the training was completed Fig. 16, comparing results of all trained models Table 3 showed that yolov7-tiny model was best among all these, other yolov7 models can also be fine-tuned to achieve better results. The final model which is yolov7-tiny, is deployed on the cloud.

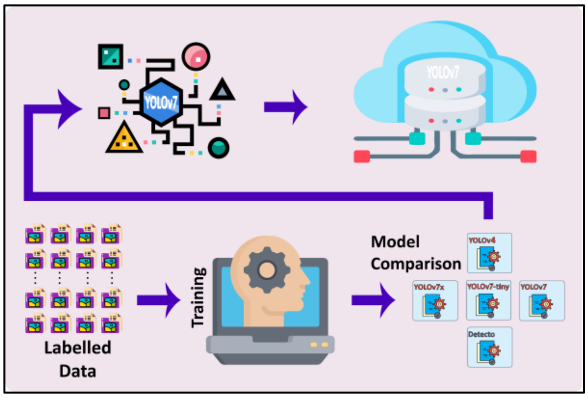

Figure 16 Model Training and Comparison

We trained several object detection models from YOLO object detection family and Detecto, Yolo family include Yolov4, Yolov7, Yolov7x, and Yolov7-tiny. The training was done on NVidia GTX 1050 Ti 4GB GPU and Google Colab [32]. The training results were then compared and we found that yolov7 has better overall performance even though yolov4 had a mAP of 100% yet its inference time was highest among all of the models

Table. 3 shows the comparison of different detection models that we used, without transformation to ONNX.

The yolov7 model is then transformed to ONNX (Open Neural Network Exchange) and is simplified which further improved inference time and the ONNX model has inference time of 0.04 seconds and that too on CPU (Intel(R) Xeon(R) Gold 5122 CPU @ 3.60GHz 3.59 GHz).

Table 3: Model Comparison

Models	mAP	Inference time	FPS
Yolov7 Tiny	0.9946	0.06 seconds	35 (On Average)
Yolov7x	0.7833	0.18 seconds	29 (On Average)
Yolov7	0.8347	0.21 seconds	19 (On Average)
Yolov4	1.0	6.00 seconds	4 (On Average)
Detecto	0.94	4.24 seconds	6 (On Average)

Based on comparison results and multiple test runs, we decided to deploy yolov7-tiny object detection model on cloud as our default model for detecting earlier defined classes, then use them to extract muzzle pattern features, identify the animal, and finally decide whether the claim for livestock insurance was fraudulent or not based on identification results.

d. Confusion matrices

The individual class confusion matrices of yolov7-tiny model are described as follows and these confusion matrices defines and summarizes the detection model performance [33]. These confusion matrices show performance of the model on training set, validation set, and test set.

- Face Class

Face class represents dataset of 4945 images for training, 269 for validation, and 56 test instances. Our model correctly classifies all images as shown in Table 4. Predicted results are same as actual.

- Nose Class

Nose class distributed as 3342 images for training, 181images for validation, and 39 for testing. Table 5 shows 99% training accuracy because of 34 images have been wrongly classified. While validation and testing accuracy is also 99%.

- Nose-Dirty Class

Nose-Dirty class has been inserted to enhance model performance. 2878 images allocated for training while 178 images for validation, and 38 images were allocated for testing. Table 6 shows 99% accurate training because of 33 images have been wrongly classified. Validation and testing accuracy is 99% and 100% respectively.

- Not-Cow Class

Not-Cow class added to prevent object selection other than Cow. It consists of 3298 images for training while 158 images for validation. Table 7 shows 100% accuracy for training and validation, and 99% for testing.

Table 4: Confusion Matrix - Class = Face

No. of Samples = 4945 (Train)	True (Train)		No. of Samples = 269 (Val)	True (Val)		No. of Samples = 56 (Test)	True (Test)	
Predicted (Train)	TP=4945	FP=0	Predicted (Val)	TP=269	FP=0	Predicted (Test)	TP=56	FP=0
	FN=0	TN=0		FN=0	TN=0		FN=0	TN=0

Table 5: Confusion Matrix - Class = Nose

No. of Samples = 3342 (Train)	True (Train)		No. of Samples = 181 (Val)	True (Val)		No. of Samples = 39 (Test)	True (Test)	
Predicted (Train)	TP=3323	FP=19	Predicted (Val)	TP=180	FP=1	Predicted (Test)	TP=38	FP=1
	FN=0	TN=0		FN=0	TN=0		FN=0	TN=0

Table 6: Confusion Matrix - Class = Nose-Dirty

No. of Samples = 2878 (Train)	True (Train)		No. of Samples = 178 (Val)	True (Val)		No. of Samples = 38 (Test)	True (Test)	
Predicted (Train)	TP=2849	FP=29	Predicted (Val)	TP=177	FP=1	Predicted (Test)	TP=38	FP=0
	FN=0	TN=0		FN=0	TN=0		FN=0	TN=0

Table 7: Confusion Matrix - Class = Not-Cow

No. of Samples = 3298 (Train)	True (Train)		No. of Samples = 158 (Val)	True (Val)		No. of Samples = 34 (Test)	True (Test)	
Predicted (Train)	TP=3298	FP=0	Predicted (Val)	TP=158	FP=0	Predicted (Test)	TP=33	FP=0
	FN=0	TN=0		FN=0	TN=0		FN=0	TN=1

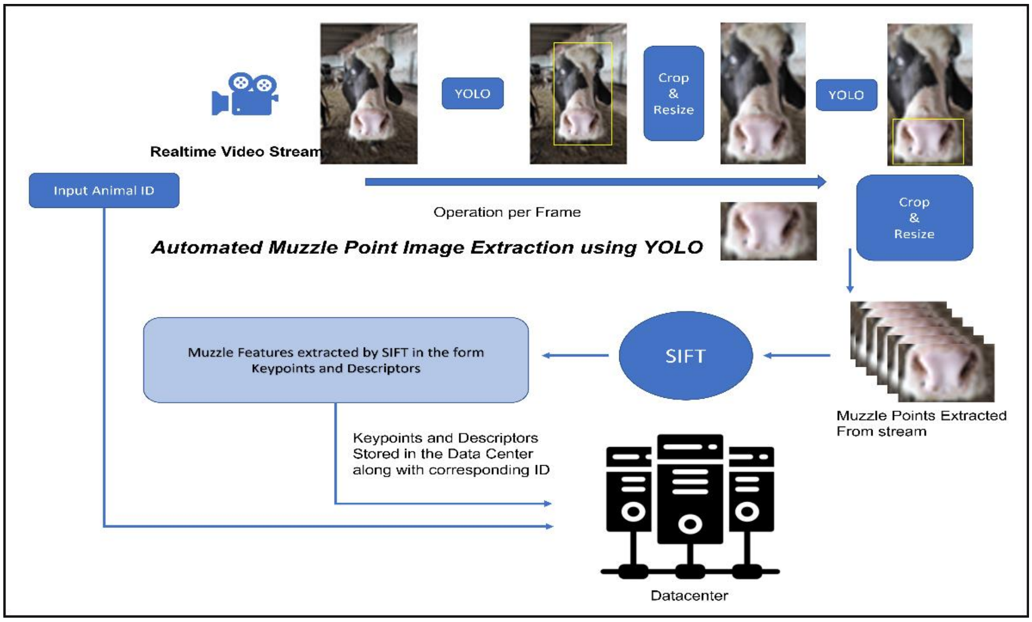

Figure 17 Animal Registration Algorithm

e. Cattle Identification Process

In our proposed model, the cattle Identification process comprises two stages. First to register the animal Table 8 & Fig 17 with specified details such as identity tag and muzzle point feature images and secondly how the system will process pattern matching for subsequent recognition of the same Animal. These processes are elaborated as follows:

1. **Animal Registration**

Table 8: Algorithm for Animal Registration

Algorithm 1 (Animal Registration):

Input:
TAG = Animal TAG;
stream = Video Stream of Animal / image of animal

Output:
image = Extracted muzzle point of Animal
stored in database with TAG

while stream **do**:
    validate(TAG)
    frame ← stream.frame
    frame ← frame.reshape
    face ← Detect(frame, "Face")
    nose ← Detect(face, "Nose")
    nose ← ExtractFeature(nose)
    image = WriteDB(nose, TAG)
    display(image)
end

a. Insertion for Animal Registration Tag

Animal tagging is the First step of the Pseudocode statement. Unique Identification string to be created for each animal. The *Registration T*ag provides a referential key to be mapped with the visual for subsequent retrieval of identity information.

b. Video stream or Images

Animal image is an essential parameter to be entered into the database. The system will extract biometric features from the images. it can be either a frame from a video or a single image.

c. Tag Validation

Before starting up the animal registration process, the system validates the tag. Duplicate, unspecified, and the substandard tag will be rejected.

d. Image reshapes as per defined standard parameters

Image resizing process perform after tag validation, Animal image to be optimized to make to compatible for feature extraction procedures.

e. Detect face area and mark as a face.

Animal face recognition is essential to detect face and then proceed to identify muzzle point. We use Yolov7 to perform face detection swiftly and accurately.

f. Detect nose within face area and mark as a nose.

Next to marking the face area, system detect Nose on the animal's face. The same procedure is applied as used for face detection.

g. Extract muzzle pattern from detect nose

At this stage, the system successfully detected the

Nose area. Now extract the muzzle pattern as the required biometric identifier.

h. Update database with muzzle pattern and Tag ID.

Finally, extracted muzzle pattern mapped Tag ID of the same cattle and store the information in the designated database management system. Show cattle image after a successful registration

2. **Recognizing the Animal**

Once the nose gets detected, the scale invariant feature transform algorithm (SIFT) is used to get key points and descriptors of muzzle point image, i.e., query image, and it also finds key points and descriptors of the images of nose/muzzle points of different animals stored, Table 9 & Fig 18.

Table 9: Algorithm for Animal Recognition

Input:
stream = Video Stream of Animal / image of animal

Output:
image, TAG = Recognized Animal with TAG
while stream **do**:
  frame ← stream.frame
  frame ← frame.reshape
  face ← Detect(frame, “Face”)
  nose ← Detect(face, “Nose”)
  nose ← ExtractFeature(nose)
  image, TAG = Recognize(nose)
  display(image, TAG)
end

Algorithm 2 (Animal Recognition):

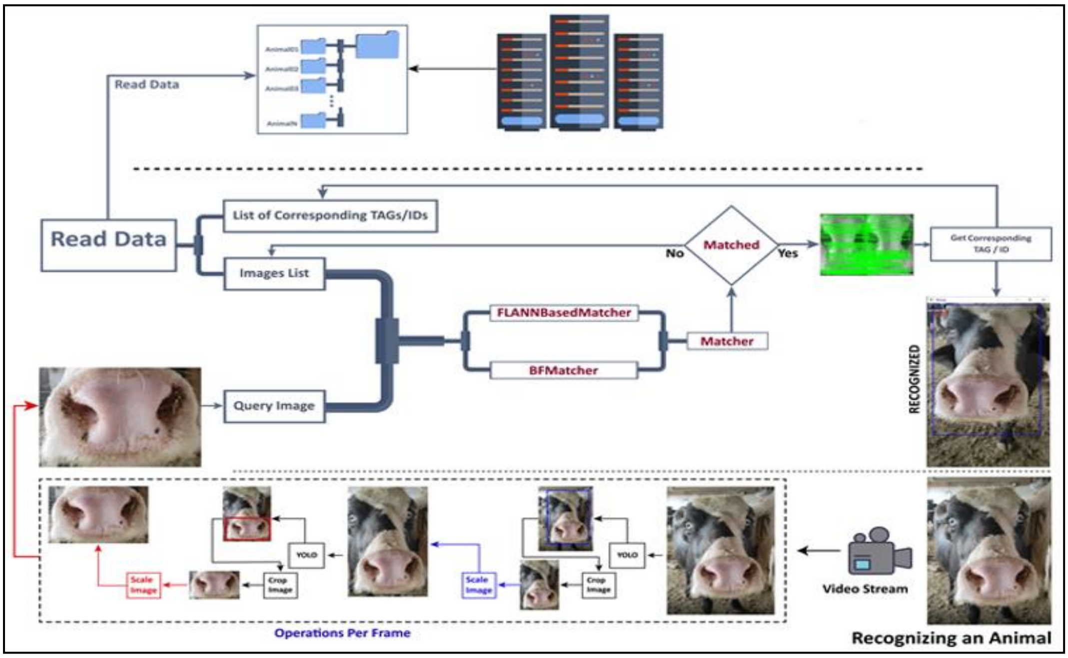

Figure 18 Animal Recognition Algorithm

3. **Scale-Invariant Feature Transform (SIFT)**

It is commonly used to extract distinguishing features from photos, which enables accurate object detection and image matching. In order to find key points—areas of the image that are unaffected by adjustments to scale, rotation, and illumination—SIFT examines image features.

a) Focus Localization

To precisely locate the key points in the image, the SIFT algorithm's key-point localization is used. It is performed by examining the image's scale-space representation, which is created by applying a number of scale-space transformations. The image is scaled to several octaves, each of which is half the size of the preceding octave, in order to locate the potential feature sites. Within each octave, a scale-space pyramid is constructed by convolving the images with Gaussian blur filters. The blurred images are represented as:

$$L(x, y, \sigma) = G(x, y, \sigma) * l(x, y) \quad (1)$$

Where x,y are the variable coordinates and σ is the “scale” parameter, G(x,y,σ) is the Gaussian function, and I(x,y) is the original image.

Next, the adjacent blurred images inside each octave are subtracted to create the Difference of Gaussians (DoG) images. The local scale-space extrema, which correlate to possible key locations, are shown by the DoG pictures.

The programme employs additional criteria to precisely localise the critical points. This includes eliminating extrema or low contrast key points, as well as key-point localization triggered by rejecting files and edges. According to the equation, files and edges in the context of SIFT correspond to particular properties that must be disregarded during key-point localization:

$$| D\ (\hat{x})\ < 0.03 \tag{2}$$

Furthermore, edges and unstable key points are rejected by analyzing the eigenvalues of the Hessian matrix H. Letting α be the larger eigenvalue and β be the smaller eigenvalue, the trace and determinant of H are calculated as:

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta \tag{3}$$

$$Det(H) = D_{xx}\, D_{yy} - \left(D_{xy}\right)^2 = \alpha\beta \tag{4}$$

Let r = α/β

So, α= rβ

$$\frac{Tr\ (H^2)}{Det\ (H)} = \frac{(\alpha+\beta)^2}{\alpha\beta} = \frac{(r\beta+\beta)^2}{r\beta^2} = \frac{(r+1)^2}{r} \tag{5}$$

The algorithm checks to see if the ratio r is smaller than 10 before computing it as r=/. Key points are disregarded if this requirement is not met when the eigenvalues are almost equal, which denotes a flat region.

The next step in the SIFT method is orientation assignment after each key point's scale, location, and orientation have been established. The algorithm now makes sure that the extracted characteristics are rotation-invariant as well as scale-invariant. To achieve this invariance, an orientation is allocated to each key point.

Each key point is given an orientation by SIFT after it has examined the local image region around it. The algorithm computes gradient magnitudes and orientations of the image pixels within this region. The dominant orientation in the region is then identified, and the key point is assigned that orientation. This step ensures that the subsequent computations are rotationally invariant.

b) Computational Cost

The scale-space pyramid construction, DoG computation, and eigenvalue analysis used in the SIFT algorithm's key-point localization step add significant processing cost. However, this cost is required to guarantee the reliability and exactness of the essential points that are discovered.

It is pertinent to remember that the computational cost of key-point localization is influenced by the size and complexity of the input image as well as the desired degree of scale-space representation. More octaves and scales are needed for huge photos, which increases computing cost. SIFT is now computationally practical for a variety of real-world applications thanks to developments in technology and optimization approaches.

c) Orientation Assignment and Key-point Descriptor

The SIFT algorithm assigns an orientation to each key point after key-point localization, making each key point rotation invariant. The algorithm's ability to match important spots in various orientations is improved by this orientation assignment.

The SIFT algorithm then computes a descriptor for each key point after determining the scale, location, and orientation of each key point. The key point's immediate surroundings are captured by the descriptor, which results in a highly distinctive depiction that is resistant to variations in scale, rotation, and illumination.

A region surrounding each key point is specified, often in the form of a square or circular neighborhood, in order to calculate the key-point descriptor. A histogram of the gradient orientations is produced within each of the smaller sub regions or bins that make up this region. The gradient orientations reveal details about the boundaries and texture of the local image structure.

Each pixel within the sub regions has its gradient magnitudes and orientations calculated, and these values are weighted by a Gaussian function centered at the critical point. This weighting prioritizes the gradients nearer the focal point, obtaining the most pertinent local data.

The weighted gradient magnitudes are then added together into orientation bins to create the histogram of gradient orientations. The key-point descriptor is formed from this histogram and is often displayed as a high-dimensional feature vector. The number of bins utilized in the orientation histogram and other factors set during the descriptor computation determine how dimensional the descriptor is.

The resulting key-point descriptors are strong and distinctive, allowing for quick matching and scene identification. To determine which critical points in distinct photos match up best, they can be compared using a variety of distance measures, such as Euclidean distance or cosine similarity.

d) FLANN Based Matcher

FLANN-based matcher or Brute Force matcher is then used to match the key points and descriptors of the query image with the images in the data, and if it matches with the image of the animal in the data, it will return the tag associated with that matched image in data, thus, recognizing/identifying the animal.

i. Video stream or Images

On the spot image/video capture to identify registered cattle.

ii. Reshape as per defined standard parameters

Same procedures have been applied to the captured image as already performed during the registration process.

iii. Match image with stored muzzle pattern and Tag ID.

Procedures are applied to detect the face and nose area to get the latest muzzle pattern. The system will look up the muzzle pattern with the stored information in the database. In case the pattern is matched, the system will get the concerned Tag ID.

4. Parameter settings

A comprehensive account of the parameter settings employed throughout our methodology.

i. For YOLOv7, the following parameters were utilized:

Input image size: 640x640
Learning rate: 0.0001
Batch size: 16
Number of epochs: 300
Optimizer: Adam
Loss function: Cross-entropy

ii. Regarding the SIFT algorithm, the following parameters were utilized:

Number of octaves: 4
Step size: 1
Contrast threshold: 0.03 (Kept to default)
Edge threshold: 10
RANSAC threshold: 3

iii. For FLANN (Fast Library for Approximate Nearest Neighbors), the following parameters were used:

K: 7
Tree type: K-D tree
Number of trees: 6
Distance metric: Euclidean distance

We recognize the importance of reporting all parameter configurations to ensure transparency and reproducibility of our methodology. By providing these details, we aim to enhance the comprehensiveness and accuracy of our paper.

4. Results ad Discussion

The best options and combinations of the strategies have been carefully evaluated for the proposed plan. The requirement for an automated and non-intrusive method to capture the muzzle point of the cattle served as the driving force for the choice of YOLOv7 as the object identification framework. Other conventional techniques, including personally photographing animals up close or employing restraints, can be time-consuming, labor-intensive, and possibly upsetting to the animals. We were able to record the animals from a safe distance while minimizing disruption to their natural behavior by utilizing YOLOv7 to automate the procedure of finding the muzzle point.

Additionally, the difficulties with other well-known techniques like CNN, RCNN, RESNET, and VGG16 that did not deliver adequate outcomes in terms of muzzle pattern identification led to the choice to use Scale-Invariant Feature Transform (SIFT) for muzzle pattern feature extraction. For extracting distinguishing features from photos, particularly when there are variations in scale, rotation, and viewpoint, SIFT is a well-known and reliable technique. We deployed SIFT to the muzzle region in an effort to record distinctive patterns that could be used for precise animal identification and recognition.

Our system can detect face and muzzle point of cow/buffalo with mAP of 99%, not only that but this system has the capability to differentiate cows/buffalos from other cattle as well as humans. The inference time is also remarkable as we have used the state of the art yolov7 object detection model. As far as recognition is concerned, the feature matching algorithm FLANN is used for muzzle point pattern recognition. The system was able to recognize the animal with 100% accuracy. The testing was done by registering the animal in the system using an image and then tested with different images in different environments. Images of a total of 500 animals were used to evaluate recognition algorithm and it recognized all the animals with 100% accuracy beating the humans because at one point we were not able to recognize the animal by just looking at the picture but the system successfully recognized the animal.

1.1. Proposed Model

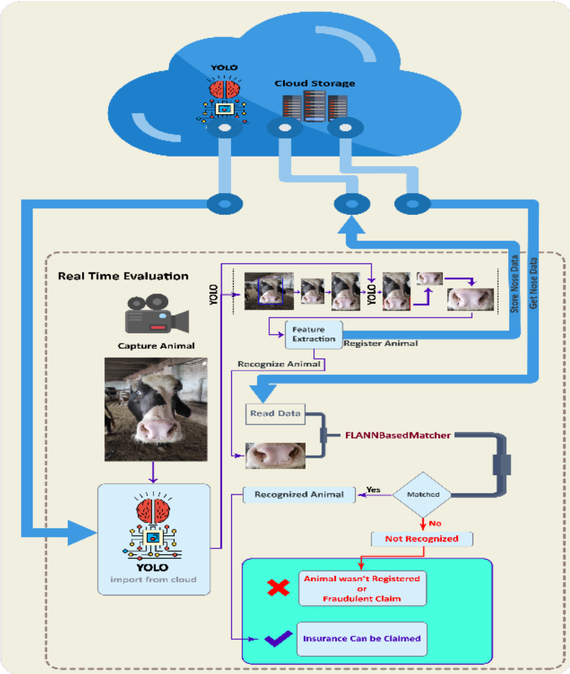

Figure 19: Proposed Model

The Fig 19 represents our proposed model for the purpose of tackling livestock insurance fraud by biometrically identifying cattle through their muzzle point patterns.

1.1. Confusion Matrix

In this research study, though our prime objective is to present a most efficient and effective model for cattle.

Identification it would be helpful for biometric identification of other Animals having muzzle pattern. Fig 20 is the confusion matrix showing the performance of the system in identification of the animal.

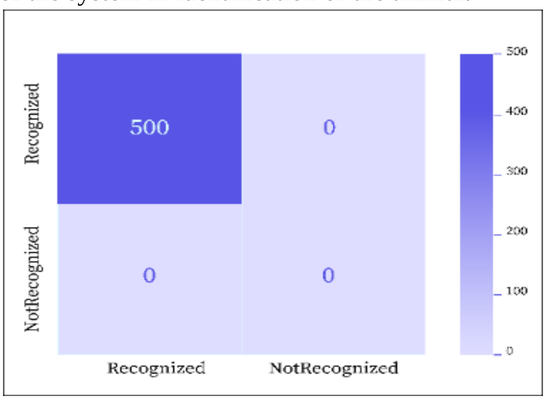

Figure 20 Cattle Identification Confusion Matrix

Following is the Table 10, enlisting animals that have unique distinguishable features:

Table 10: Animal Unique Features

Sr. No	Animal	Unique Feature
1	Cows	Muzzle/Nose Pattern
2	Bulls	Muzzle/Nose Pattern
3	Calves	Muzzle/Nose Pattern
4	Buffalos	Muzzle/Nose Pattern
5	Horses	Muzzle/Nose Pattern
6	Goats	Muzzle/Nose Pattern
7	Sheep	Muzzle/Nose Patter
8	Cats	Nose Pattern
9	Dogs	Nose Pattern
10	Lions	Whisker Holes

Our proposed model takes cattle images/videos in real-time and yolo object detection model extracts muzzle point in two steps, first is to extract the head/face of the animal in order to make sure that only cow, buffalo, bull, and/or a calf is present in the scene, next once the muzzle point is taken out, its features are extracted and stores in case of animal registration. In case of recognition, the extracted features are then transferred to the matcher algorithm which matches the muzzle point features with already stored muzzle features in database and in case of a hit, it returns the tag of the animal which can assure genuineness of the insurance claim, but if the matcher does not recognize, then that would mean that the animal was not registered in the system which subsequently tells that the claim for livestock insurance was fraudulent.

The algorithms for cattle identification systems are compared in Table 11, along with the detection and identification accuracy rates. The first system uses the Fuzzy-KNN algorithm in combination with the Hybrid Feature Extraction and Classification approach to achieve an accuracy rate of 96.74% with a loss rate of 3.26%. However, as animals tend to flee, collecting muzzle point photographs of cattle for this technique can be difficult.

The second system achieves a 95.13% accuracy rate with a 4.87% loss rate using the BoHoG and LBP Histogram algorithms. However, they use the watershed approach to estimate the location of the muzzle point in the image's center, which might not be precise in real-time scenarios.

Table 10: Comparing Proposed Model with Different Research Approaches

Sr .No	Title	Algorithms	Detection Accuracy Rate (%)	Detection Loss Rate (%)	Identification Accuracy Rate (%)	Identification Loss Rate (%)
1	Hybrid Feature Extraction and Classification	Fuzzy-KNN	-	-	96.74	3.26
2	Biometric for Cattle Identification using Muzzle Patterns	BoHoG and LBP Histogram	-	-	95.13	4.87
3	Cattle Face Recognition Method Based on Parameter Transfer and Deep Learning	VGGNet-DCNN	-	-	70.00	30.00
4	Cow Identification System Using Ear Tag Recognition	YOLO	96.00	4.00	-	-
5	Automated Individual Cattle Identification Using Video Data: A Unified Deep Learning Architecture Approach	BiLSTM. Inception-V3 CNN	-	-	93.30	6.70
6	Automated Muzzle Detection and Biometric Identification	YOLO & Few-Shot Deep Transfer Learning	99.11	0.99	-	-

With the help of the VGGNet-DCNN algorithm and the Cattle Face Recognition Method, the third system achieves an identification accuracy rate of 70.00%. The YOLO algorithm, which is used by the fourth system, has a detection accuracy rate of 96.00%. However, because they can be switched or faked, ear tags are not a reliable method for identifying cows.

The fifth system achieves a 93.30% identification accuracy rate with a 6.70% loss rate by utilizing a Unified Deep Learning Architecture method with BiLSTM and Inception-V3 CNN algorithms. However, after a certain frame length, the accuracy based on video frame length may not improve or even decline.

The sixth system uses YOLO and Few-Shot Deep Transfer Learning algorithms for Automated Muzzle Detection and Biometric Identification, achieving a 99.11% detection accuracy rate with a 0.99% loss rate.

Our proposed algorithm, which uses YOLOV7 and Muzzle Pattern Feature Matching algorithms, achieved exceptional results. It achieved a detection accuracy rate of 99.50% with a 0.50% loss rate and a perfect identification accuracy rate of 100.00% with 0.00% loss rate. The proposed deep learning-based systems using YOLO algorithm demonstrated superior performance in terms of detection and identification accuracy rates.

5. Conclusion

Accurate identification of cattle is very critical and remains challenging for veterinarian and livestock Insurer. They have been facing problems for registration and subsequent accurate recognition of registered animals. Livestock insurance frauds not only entail financial losses to the insurers but they also

limit the capacity of livestock Insurers to provide services to the potential clients.

In this paper, we introduced a unique approach based on Yolov7 techniques of object detection. It detects objects more accurately and swiftly than the previous versions. Our dataset includes 9400 images. Total annotated dataset comprises 15,416 class labels representing all the four classes face, nose, dirty nose and not cow. Then had to be split into training ~94%, validation ~5%, and testing 1%

Images of a total of 500 animals were used to evaluate the recognition algorithm and it recognized all the animals with 100% accuracy as we have kept the threshold of recognition by considering the sensitivity of the risk involved Animal is considered as recognized if the recognition algorithm matches at most 10 key points and descriptors. But this can be changed which can change the accuracy of animal identification.

The proposed model may be applied to resolve cattle identification current issues, and also lead to exploring other identification technologies to protect livestock from serious diseases and avoid financial loss. Likewise, it may be helpful for Livestock Insurers to substantiate valid claims by accurate identification of the exactly insured animal. We are confident that this research work will provide a firm basis for future studies on the identification of other animals having muzzle patterns as listed in Table 10. The application of more advanced methodologies with an extended system for a huge dataset is highly recommended as a research area for future work. Researchers should focus on techniques to provide more accurate robust identification irrespective of the muzzle pattern quality.

Funding Statement: This research program is partially funded by the United Insurance Company of Pakistan Ltd. in collaboration with United Software and Technologies International (Pvt) Ltd. under Grant No.UICL-06-2022/USTI-RND-006/11/2022.

References

[1] Awad AI, Hassaballah M. Bag-of-visual-words for cattle identification from muzzle print images. Appl Sci 2019;9:4914.

[2] Li G, Erickson GE, Xiong Y. Individual Beef Cattle Identification Using Muzzle Images and Deep Learning Techniques. Animals 2022;12:1453.

[3] Awad AI, Hassanien AE, Zawbaa HM. A cattle identification approach using live captured muzzle print images. Adv. Secur. Inf. Commun. Networks First Int. Conf. SecNet 2013, Cairo, Egypt, Sept. 3-5, 2013. Proc., 2013, p. 143–52.

[4] Kumar S, Pandey A, Satwik KSR, Kumar S, Singh SK, Singh AK, et al. Deep learning framework for recognition of cattle using muzzle point image pattern. Measurement 2018;116:1–17.

[5] Tharwat A, Gaber T, Hassanien AE. Two biometric approaches for cattle identification based on features and classifiers fusion. Int J Image Min 2015;1:342–65.

[6] Ammour B, Boubchir L, Bouden T, Ramdani M. Face--iris multimodal biometric identification system. Electronics 2020;9:85.

[7] Neethirajan S, Reimert I, Kemp B. Measuring farm animal emotions—Sensor-based approaches. Sensors 2021;21:553.

[8] Kumar S, Singh SK. Visual animal biometrics: survey. IET Biometrics 2017;6:139–56.

[9] Kumar S, Singh SK, Abidi AI, Datta D, Sangaiah AK. Group sparse representation approach for recognition of cattle on muzzle point images. Int J Parallel Program 2018;46:812–37.

[10] Kumar S, Singh SK, Singh AK. Muzzle point pattern based techniques for individual cattle identification. IET Image Process 2017;11:805–14.

[11] Bello RW, Olubummo DA, Seiyaboh Z, Enuma OC, Talib AZ, Mohamed ASA. Cattle identification: the history of nose prints approach in brief. IOP Conf. Ser. Earth Environ. Sci., vol. 594, 2020, p. 12026.

[12] Kumar S, Singh SK. Automatic identification of cattle using muzzle point pattern: a hybrid feature extraction and classification paradigm. Multimed Tools Appl 2017;76:26551–80.

[13] Ruhil AP, Mohanty TK, Rao SVN, Lathwal SS, Subramanian VV, others. Radio-frequency identification: A cost effective tool to improve livestock sector. Indian J Anim Sci 2013;83:871–9.

[14] Garcia R, Aguilar J, Toro M, Pinto A, Rodriguez P. A systematic literature review on the use of machine learning in precision livestock farming. Comput Electron Agric 2020;179:105826.

[15] Xu B, Wang W, Guo L, Chen G, Wang Y, Zhang W, et al. Evaluation of deep learning for automatic multi-view face detection in cattle. Agriculture 2021;11:1062.

[16] Awad AI. From classical methods to animal biometrics: A review on cattle identification and tracking. Comput Electron Agric 2016;123:423–35.

[17] Herlin A, Brunberg E, Hultgren J, Högberg N, Rydberg A, Skarin A. Animal welfare implications of digital tools for monitoring and management of cattle and sheep on pasture. Animals 2021;11:829.

[18] Ahmed S, Gaber T, Tharwat A, Hassanien AE, Snáel V. Muzzle-based cattle identification using speed up robust feature approach. 2015 Int. Conf. Intell. Netw. Collab. Syst., 2015, p. 99–104.

[19] Kusakunniran W, Wiratsudakul A, Chuachan U, Kanchanapreechakorn S, Imaromkul T. Automatic cattle identification based on fusion of texture features extracted from muzzle images. 2018 IEEE Int. Conf. Ind. Technol., 2018, p. 1484–9.

[20] Rudenko O, Megel Y, Bezsonov O, Rybalka A. Cattle breed identification and live weight evaluation on the basis of machine learning and computer vision. CMIS, 2020, p. 939–54.

[21] Kumar S, Singh SK, Singh R, Singh AK. Animal biometrics: Techniques and applications. Anim Biometrics Tech Appl 2018:1–243. https://doi.org/10.1007/978-981-10-7956-6.

[22] Adelson EH, Anderson CH, Bergen JR, Burt PJ, Ogden JM. Pyramid methods in image processing. RCA Eng 1984;29:33–41.

[23] Zin TT, Misawa S, Pwint MZ, Thant S, Seint PT, Sumi K, et al. Cow identification system using ear tag recognition. 2020 IEEE 2nd Glob. Conf. Life Sci. Technol., 2020, p. 65–6.

[24] Bochkovskiy A, Wang C-Y, Liao H-YM. Yolov4: Optimal speed and accuracy of object detection. ArXiv Prepr ArXiv200410934 2020.

[25] Wang H, Qin J, Hou Q, Gong S. Cattle face recognition method based on parameter transfer and deep learning. J. Phys. Conf. Ser., vol. 1453, 2020, p. 12054.

[26] Kumar S, Singh SK, Singh RS, Singh AK, Tiwari S. Real-time recognition of cattle using animal biometrics. J Real-Time Image Process 2017;13:505–26.

[27] Kumar S, Singh SK. Cattle recognition: A new frontier in visual

animal biometrics research. Proc Natl Acad Sci India Sect A Phys Sci 2020;90:689–708.
[28]Shojaeipour A, Falzon G, Kwan P, Hadavi N, Cowley FC, Paul D. Automated muzzle detection and biometric identification via few-shot deep transfer learning of mixed breed cattle. Agronomy 2021;11:2365.
[29]Tzutalin. labelImg. Https://GithubCom/ 2018.
[30]Bi A. Detecto 2019.
[31]Yang F, Zhang X, Liu B. Video object tracking based on YOLOv7 and DeepSORT 2022;14:2015–8.
[32]Google. Google Colab. Https://ColabResearchGoogleCom/ n.d.
[33]Bowes D, Hall T, Gray D. Comparing the performance of fault prediction models which report multiple performance measures: Recomputing the confusion matrix. ACM Int Conf Proceeding Ser 2012:109–18. https://doi.org/10.1145/2365324.2365338.