Final report

Cattle counting and weighing AI

Project code	LC.RDE.0019
Prepared by	Gergely Boromisza Andras Karpinszky Dr Julia Beata Pinter
Date published	5 December 2024
Published for	LiveCorp PO Box 1174 NORTH SYDNEY NSW 2059
	Meat & Livestock Australia Limited Locked Bag 991 NORTH SYDNEY NSW 2059
In partnership with	PigBrother Kft.

LiveCorp and Meat & Livestock Australia acknowledge the contribution from the Commonwealth of Australia to research and development undertaken in the LEP RD&E Program.

This document is published by the Livestock Export Program (LEP), a collaboration between the Australian Livestock Export Corporation (LiveCorp) and Meat & Livestock Australia (MLA). Care is taken to ensure the accuracy of the information contained in this document. However, LiveCorp and MLA cannot accept responsibility for the accuracy or completeness of the information or opinions contained in the document. No person should act on the basis of the contents of this publication without first obtaining specific, independent professional advice. Any recommendations, suggestions or opinions contained in this publication do not necessarily represent the policy or views of LiveCorp and MLA. Reproduction in whole or in part of this publication is prohibited without prior written consent from LiveCorp.

1. Executive summary

The project aimed to adapt an advanced visual recognition system, originally developed for other livestock industries, for use in livestock export facilities. Its purpose was to trial the application of artificial intelligence (AI) in automating cattle counting and weight estimation (or percentage change). The project commenced with a visit to a livestock export ship, followed by a site visit to map the facility in detail and deploy cameras, enabling data collection and AI training to begin.

For the CattleCounter solution, high accuracy was achieved using an enclosed cage with stable lighting and a camera positioned above the animals. While accuracy was improved over the earlier testing by adjusting the front-facing camera, further testing is required to fully evaluate the system's performance.

For the CattleWeigherEstimator, both RGB and 3D data was used to ensure accurate and reliable weight estimation. This approach assessed the strengths and limitations of each data type and optimise the model. Three models were trained for weight estimation using different cameras: RGB approach, RGB cross-camera validation approach, and RGB cross-camera validation grayscale approach.

Overall, the analysis showed that both categories are significant at the 95% confidence level, with the weight category also significant at the 98% confidence level. Further testing is recommended to continually improve the AI model.

Transparency, digitisation, and animal welfare are key objectives in Australian agriculture. PigBrother Ltd. will approach interested companies regarding the AI-based cattle counting and weighing technologies tested in this proof of concept project.

Table of contents

List of Figures

2. Site visits

To prepare for data collection, fieldwork commenced with a visit to a livestock export ship on 15 October 2023 (see Fig. 1). The visit included observing and inspecting the entire process of loading cattle onto the ship and assessing the feasibility of installing a camera system for counting and weighing the cattle. During the visit, the following locations were highlighted:

- The location where cattle are unloaded from the trucks.
- The area where cattle move through the race and onto the ship.
- The spaces where cattle are housed for the duration of their journey.

Figure 1: PigBrother team and LiveCorp representative on the ship visit

As a result of inspecting the livestock loading process, the following initial observations were made regarding an installation on a livestock export ship:

- The on-board computer for the counting and weight estimation system should be placed in a clean environment to protect it from potential contamination or adverse environmental conditions. To achieve this, a cable would need to be routed between the camera and the computer. The feasibility of a mobile computer system was also considered; however, further exploration is required to address questions about usability, transportability, and durability.
- The initial assessment indicated that counting livestock as they are loaded onto the ship is feasible. However, further collaboration with the ship's crew is necessary to refine installation methods. Additionally, the availability of Wi-Fi coverage on board was evaluated. A Wi-Fi access point may need to be installed near the measurement area to enable system access, such as via a mobile device.
- Research is ongoing regarding the implementation of weight estimation. Efforts are focused on achieving greater accuracy using either traditional RGB cameras or 3D cameras. The environment appears suitable for measurements, with relatively stable lighting and cattle passing through a narrow space in single file rather than side by side.
- Technical support for the system's installation on board was offered; however, this falls outside the scope of the current task and requires further discussion.

The proof of concept trial was conducted at a cattle feedlot in Broome (see Fig. 2). The initial investigation focused on identifying key location points where cattle movement occurred, such as the feedlot's entry and exit points. At each of these points, an assessment was carried out to determine the optimal installation of cameras and the most suitable positioning for the processing unit. This evaluation also considered the influence of lighting conditions on the quality of the camera images.

Figure 2: Feedlot survey in Broome

3. Installation & maintenance of cameras for counting and weighing solutions

3.1 Data collection

Data collection and AI model training focused on red Brahman, white Brahman, Droughtmaster, and crossbreed cattle. During data collection, several types of footage were gathered to enhance understanding and model training:

- RGB camera videos from a top view, capturing cattle mostly moving one by one
- RGB camera videos from a top view, capturing cattle arranged in groups
- RGB camera videos from a front view, capturing cattle moving one by one
- RGB camera videos from an eye-level view under night conditions with illumination
- RGBD camera videos, capturing cattle mostly moving one by one

3.2 Currently applied weighing procedure and data collection considerations

As a preliminary step, to assess and select the most optimal camera placement points from the project's perspective, it was crucial to get an in-depth understanding of the process of weighing cattle at the feedlot.

The weighing process follows the steps below:

1. Approximately 10-20 cattle are brought into the collection area.
2. The cattle enter one by one into this area and wait at the gate.
3. The operator, by opening and closing the gate, allows each animal to enter the area in front of the scale. Typically, only one animal may be present here at a time.

4. When the scale becomes available, the operator, using control mechanisms, opens the door of the scale, allowing the waiting cattle in and then closes the door (this is labelled as “cattle crush”). The RFID tag of the cattle is read, its weight is measured on the scale and, if necessary, other interventions (e.g. vaccination) take place. The RFID tag values appear on the operator's scale display, and the system saves them in a database. The operator opens the exit of the scale, and the animal leaves the scale.
5. The process repeats from step three.

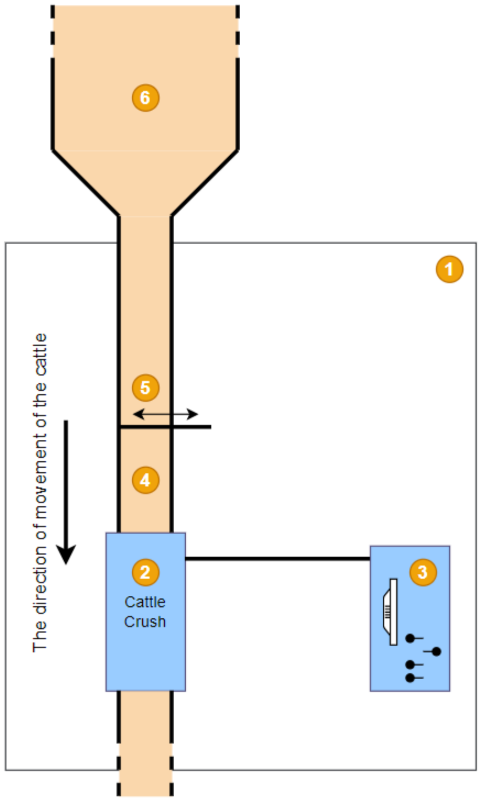

Figure 3: Outline of the cattle weighing process

1 – Weighing facility
2 – Cattle crush, scale and RFID reader
3 – Cattle crush controls and user interface
4 – Area between the cattle crush and the gate where the animal waits
5 – Area located in front of the gate where the animals form a row
6 – Collection area

3.3 Weight estimator system

The cameras were initially installed in front of the entry gate (Fig. 4), which was the optimal location. However, due to frequent cattle congestion in this area, resulting in potentially inaccurate imaging data for the weighing system, the cameras were subsequently relocated to the area behind the gate.

The narrow corridor between the gate and the scale was later identified as a more suitable location for weight data collection. This setup allowed for the capture of significantly more reliable and accurate image data (see Fig. 5). In this area, cattle image data from the footage and weight data from the cattle crush were made available in .csv format, which the team received daily. Subsequent field visits concentrated on collecting weight data from this location.

Figure 4: Temporarily mounted 3D (left) and RGB (right) cameras

Figure 5: Image from top view camera installed between the scale and the gate

3.4 Counter platform

It was observed that the cattle gathered in the collection areas, and their movements, could provide valuable data for training the counting system's model. As a result, the counting RGB camera was temporarily repositioned to this area (see Fig. 6) to capture this data.

During the transportation period, the counting system camera was relocated to the designated area where cattle were loaded onto transport vehicles. At this location, the low ceiling at the point where each animal passed posed a challenge, potentially restricting the camera's field of view. To mitigate this, the camera was mounted on a fence post at an angle, capturing a frontal and slightly side view (see Fig. 7). Data collection at this site occurred during both daytime and nighttime, allowing for a diverse dataset.

Figure 6: Image from top view camera installed above the collection area

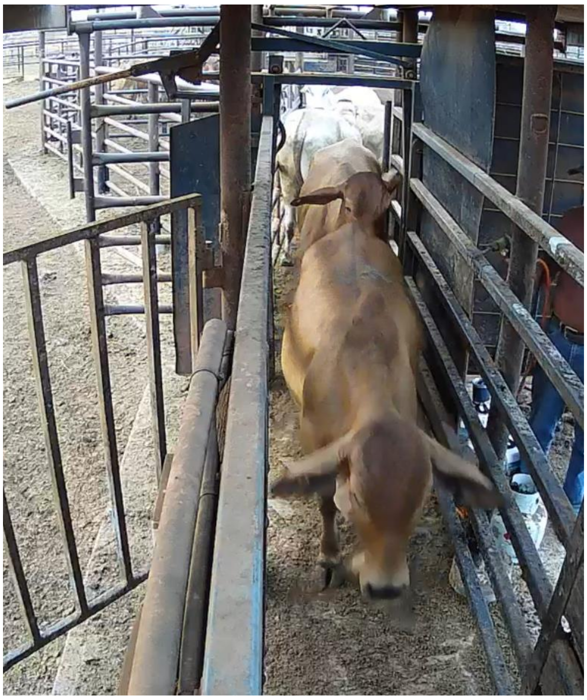

Figure 7: Frontal view from ("front view") mounted camera

4. CattleCounter solution

4.1 Testing conditions

Given the limited availability of alternative testing sites and the need to evaluate the model under varying conditions, it was decided to use recordings from multiple cameras, excluding the one used for the test dataset. Specifically, this included utilising the RGBD camera used during the weight estimation process.

Figure 8: Two different camera views

As seen in Fig.8, there is a notable disparity between the images captured by the two cameras. This dissimilarity encompasses variations in colour rendition, object size and the presence of a fisheye effect. These disparities underscore the necessity of evaluating the model's performance across diverse visual inputs to ensure robustness and accuracy in its predictions.

Another crucial factor is that the RGBD camera used for testing purposes experienced some downtime. This resulted in fewer data captures during the testing phase. Consequently, the total count of discernible objects in the test case is expected to be lower than those previously captured with the regular camera.

Considering the data loss described above, all the output test videos were meticulously reviewed to identify any potential errors and accurately count the animals passing underneath the camera. To facilitate this process, the recordings were divided into smaller segments. Each segment was defined by either a one-second gap in data or when the output video reached a duration of five minutes. This segmentation strategy not only made the testing process more manageable but also enabled the search for anomalies that might occur in case of potential system outages.

4.2 Top view results

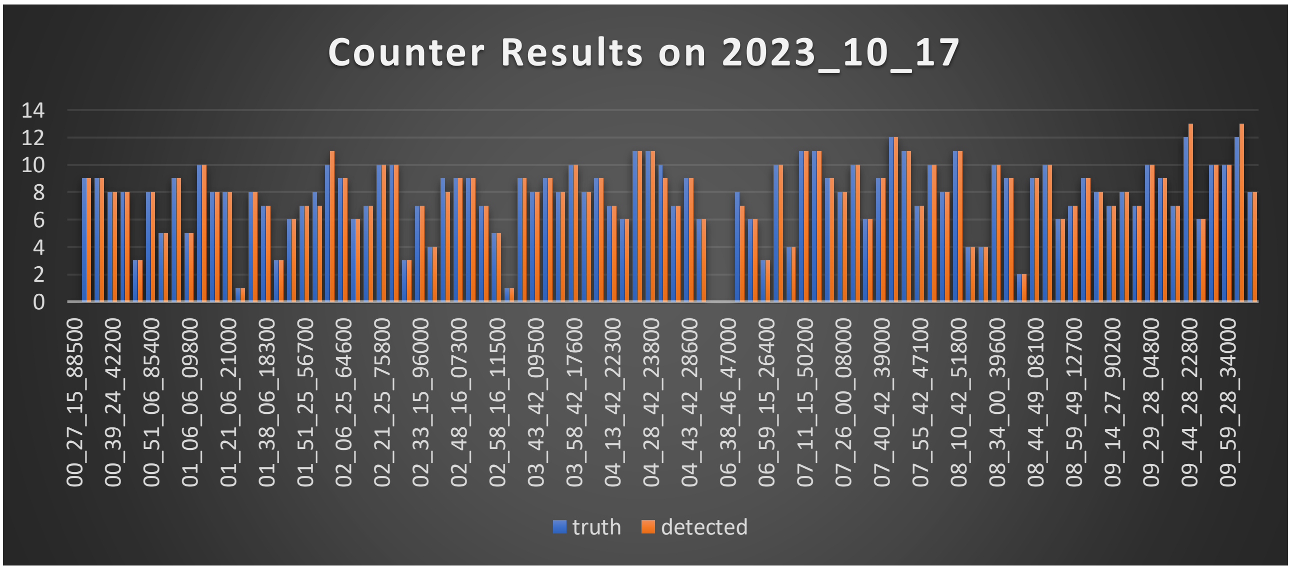

Figure 9: CattleCounter results

As shown in the chart above, 93 video segments were utilised for this evaluation. Within these segments, a total of 682 animals were identified, with seven instances of counting errors. Consequently, the achieved accuracy of 98.97% closely aligns with the early estimations of 99%.

The errors observed were mostly due to multiple animals being near each other. To achieve 100% detection accuracy even when there are multiple animals around, the research team investigated these mistakes. Interestingly, five out of the seven errors shared a common characteristic: the animal directly under the camera had its head lowered. Based on this observation, it appears that the current model places significant emphasis on the relevance of the head and utilises its presence to detect the individual animal.

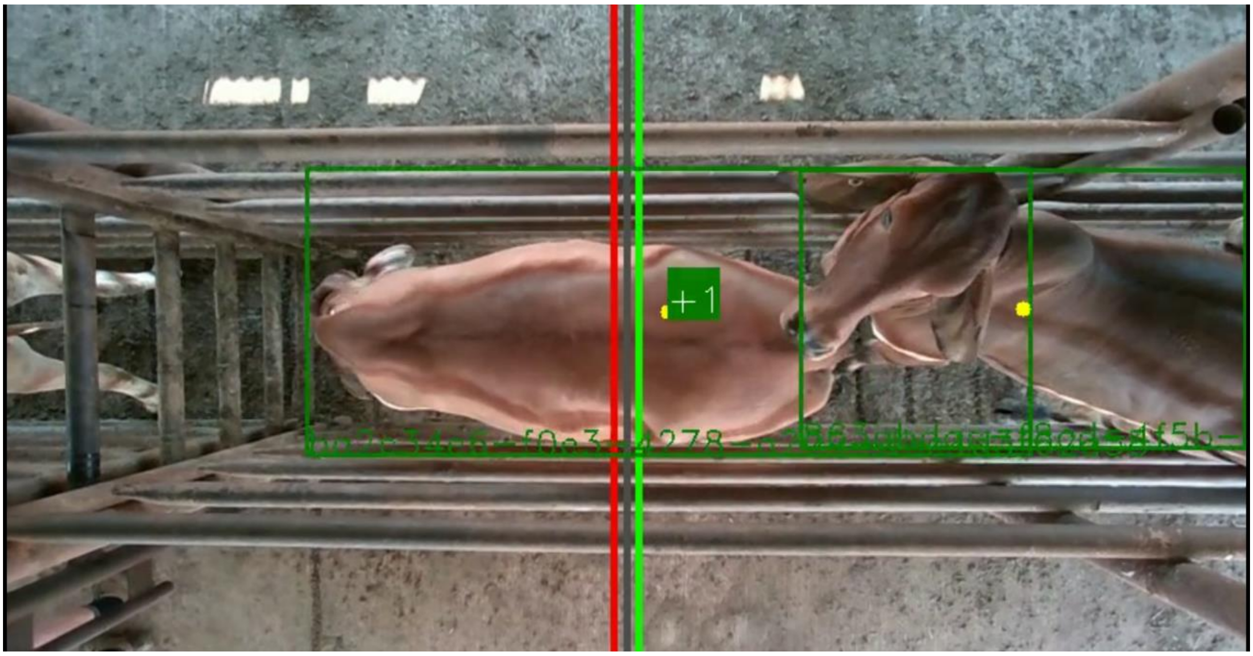

Figure 10: A picture of an error

The errors are documented and ready for annotation. Additional data to the training set will resolve these issues.

Another observation is that foreign objects, such as workers' equipment, and additional sunlight did not pose any problems in this environment. As such, it did not have an adverse impact on accuracy.

In this test case, most of the animals were red Brahmans. To ensure the versatility of the model across different colour variations of animals, additional testing was conducted on a day when white Brahmans were also present. The results were highly promising, with all 113 animals accurately detected without errors. This demonstrated the model's robust and reliable detection capabilities for both red and white Brahmans.

4.3 Front view results

The front-view setup presented several challenges, including data variability and environmental factors such as insufficient lighting and shadows cast on the animals by surrounding objects. The two-way traffic in the hallway further complicated the system's performance, as animals needed to be recognised from both the front and rear. Additionally, the presence of workers in the hallway introduced potential errors, as they were occasionally misidentified as animals.

The initial testing of the system revealed several issues that required resolution. Problematic recordings were identified, with some excluded for later testing, while others were incorporated into the training data to improve the model. Adjustments were also made to the system to enhance its suitability for this setup.

However, technical difficulties prevented the original format of the videos from being saved, leaving only the output video.

Additional work was necessary to make the output video usable for both testing and training purposes. The counting line drawn on the video had to be removed; however, remnants of the counting line and bounding boxes remain visible, as shown in the image below.

Figure 11: Image of noisy video

Despite pre-processing efforts, the output video retained noise and exhibited lower resolution compared to the original recording. Nevertheless, the evaluation was conducted, and the results are presented below.

Animal count	Old results	New result
180	3	111
180	9	x
180	19	92
180	19	48
180	84	106
180	110	93
180	144	97
180	113	95
180	11	88
180	-1	109
180	0	112
180	0	50

As depicted in the table, despite the low quality of the test video, a higher counting accuracy was achieved in most cases. However, it is important to note that this accuracy still falls short of the true value. Nevertheless, it is believed that with the right input video, even higher accuracy levels can be achieved.

4.4 Conclusions

The findings suggest that high accuracy can be achieved when the system is used in a closed enclosure with relatively stable lighting and the camera positioned above the animals. While the accuracy was higher compared to earlier tests with the front-view camera setup, additional testing is needed to gain a clearer understanding of the system's overall performance.

4.5 Graphical user interface

To enhance ease of use, the team developed a graphical user interface (GUI) that is compatible with multiple devices. The GUI is accessible from a web browser on mobile phones, tablets and computers by logging into the same network. The images below are from the application that controls the animal counter service.

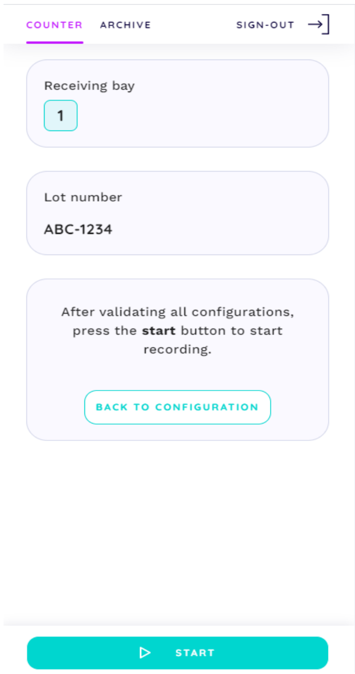

Figure 12: AnimalCounter - Screen 1

On screen 1 the user has already set up the lot number (truck id, customer name, arbitrary text) and has chosen the camera which should do the counting (there is possibility to count simultaneously on multiple cameras with the same GUI).

Figure 13: AnimalCounter – Screen 2

On screen 2 the actual count is visible. This screen refreshes itself every second.

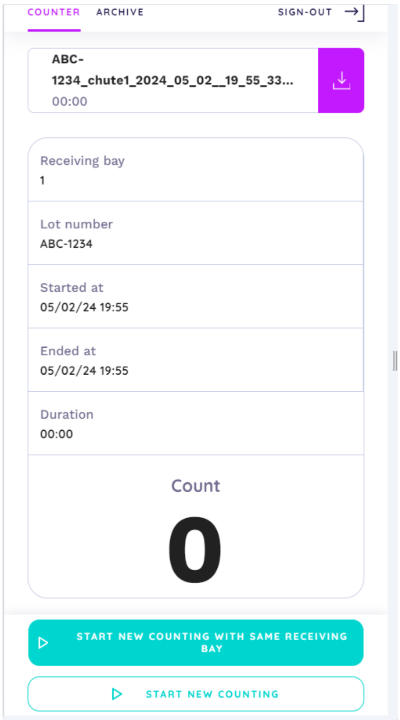

Figure 14: AnimalCounter – Screen 3

Once you have clicked on the STOP button, the counting session will stop. You will see a short summary about the counting session: start and stop date, count, lot number (screen 3). You can start a counting again directly from this screen or go to archive.

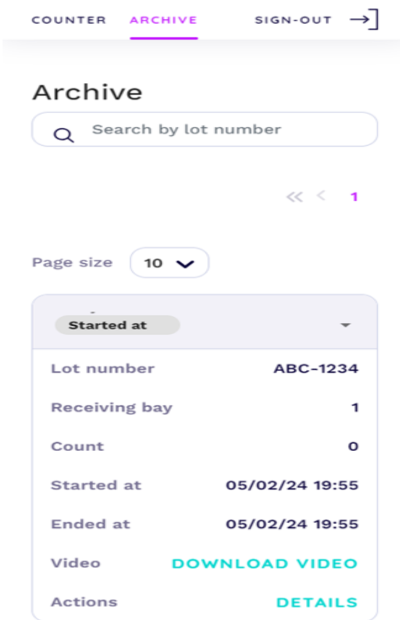

Figure 15: AnimalCounter – Screen 4

On screen 4 the Archive is visible. It contains information about the counting session, and the user can download the annotated video to a local device from this screen.

4.6 Use cases

Use cases define how end-users deploy and assess the application. Sending animals from one place to another is always a complicated task, requiring significant coordination, communication, and manual labour. Accurately counting the animals is essential, particularly during loading and unloading from trucks or vessels, where counting is typically done manually. The CattleCounter solution automates this process, reducing the need for extensive manual labour.

CattleCounter effectively addresses this challenge by:

- Counting animals with 99.8% accuracy (once retrained for the specific location).
- Creating annotated videos that include timestamps, the exact count of animals at each frame, and truck numbers.
- Offering an easy-to-use, user-friendly graphical user interface for quick and seamless control of the application.
- Allowing counting to be performed on an edge server, meaning no internet connection is required once installed.

A single server can support up to five simultaneous counting instances with five different cameras, enabling the operation of the system at two separate points to ensure the AI models perform optimally. As the cameras are independent, the likelihood of both models failing is the product of their individual error probabilities: 0.02 * 0.02 = 0.0004 or 0.04%, which is extremely low.

4.7 Installation

There is no specific camera used in the CattleCounter solution. Instead, the camera must meet several basic requirements that are common for IP security cameras:

- IP67 rating
- Power over Ethernet (PoE)
- Support for RTSP protocol
- Fixed focal length
- Infrared capability

The placement of the camera should be chosen carefully to ensure optimal performance:

- Narrow corridors are preferable over wide spaces.
- The entire animal, from head to tail, should be visible in the camera’s frame.
- The camera should be protected from rain, as raindrops (along with mist, fog, dirt, and flies) can negatively impact its vision capabilities.

Once the camera placement is chosen, the cameras should be securely fixed in place to prevent any movement during data collection, as this could affect the results. The counting process is carried out by an edge server, which should be connected to the camera via a local Wi-Fi router. This setup enables users to easily connect to the application’s GUI and control it (start, stop, and download videos). Before using the application, the following steps must be completed:

- A server should be installed on user’s site.
- Cameras should be positioned in a top-down position.

- The server and cameras should be connected to the same local LAN network.
- Remote access needs to be granted for PigBrother to install and configure the software.

The server's hardware requirements will depend on the number of cameras to be used simultaneously. This specification will be provided by PigBrother. The server should run the Ubuntu 22.04 headless version. An SSH connection is required for PigBrother to install the necessary drivers, third-party dependencies, and the latest version of the application. Once installation and accuracy verification are complete, the server can be disconnected from the public internet.

4.8 Maintenance

It is recommended to clean the camera lenses once every two weeks or after extended periods of usage. The application automatically manages the created videos, deleting those older than three weeks (this setting can be adjusted to meet the customer's needs).

4.9 How to improve accuracy

As each site is unique, overall accuracy tends to improve if a 2-4 hour video is provided from the site with animals, which can be used to retrain the model to better understand the site's specific environmental conditions. If accuracy issues arise during the use of the application, customers can contact PigBrother, providing footage of the errors, and the model will be retrained. In cases where multiple transportation instances occur at night, lighting conditions should be addressed separately to ensure optimal performance.

5. Cattle WeightEstimator solution

Two approaches were explored to investigate cattle weight estimation: one using RGB data and the other utilising 3D data. In the initial phase, RGB data was used to train and validate the model without any modifications. In the second phase, the model was again trained using RGB data, but with a focus on minimising the 'fisheye' distortion caused by the architecture of the RGB camera. This was achieved through advanced distortion correction algorithms. This step was essential since the model was validated using 3D data, which does not exhibit the same 'fisheye' distortion.

By integrating both RGB and 3D data, a comprehensive understanding of cattle weight estimation was sought to ensure the accuracy and reliability of the model's predictions. This multifaceted approach allowed for the evaluation of the strengths and limitations of each data modality, ultimately optimising the model's performance. To test the weight estimation solution with different camera types, three distinct models were trained and evaluated:

- RGB approach – this model was trained on Dahua camera data, tested on Dahua camera data.
- RGB cross-camera validation approach – model was trained on transformed (unfisheyed) Dahua camera data, tested on Intel Real sense d455 camera rgb channel.
- RGB cross-camera validation grayscale approach – model was trained on transformed (unfisheyed, grayscaled) Dahua camera data, tested on Intel Real sense d455 camera rgb channel (grayscale).

5.1 Data processing

The data processing phase was crucial for the success of the trial. Raw images were extracted from the videos, capturing key information such as electronic ID, date, and weight, which were necessary for subsequent analysis. Following this, specific filters were applied to the raw images to prepare them for further processing.

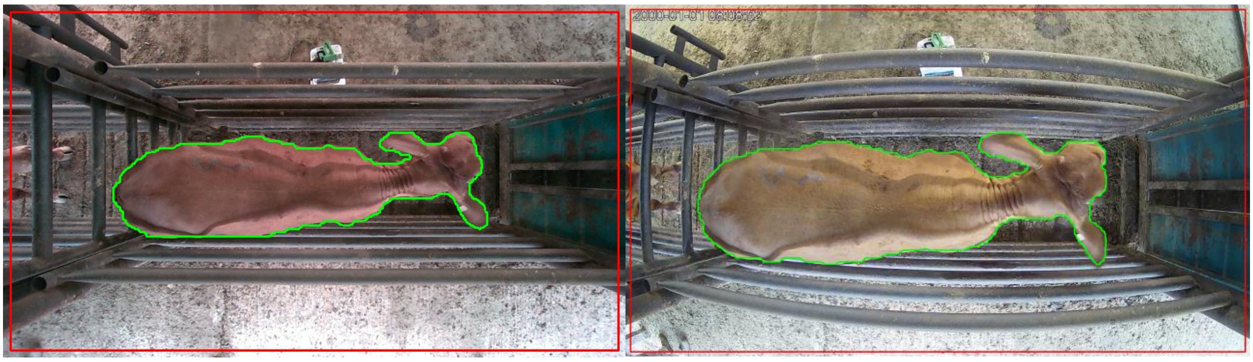

Figure 16: Left - 3D image with filter, Right - RGB image with filter

Firstly, a segmentation neural network was trained using the data collected for cattle counting. During this process, the annotations were adjusted to meet the requirements of training the model. As a result, cattle "masks" were obtained, which were then used for further processing.

Only fully visible cattle, as indicated by the red rectangle, were included in the dataset. Any animal parts outside this rectangle were excluded. Additionally, animals close to the one being recorded were also excluded. Consequently, only the correct image and mask of the animal being recorded were retained. Further modifications and filters will be discussed in the RGB, 3D, and 3D-grayscale sections.

5.2 RGB approach

In this section, the RGB approach used in the research is outlined. As mentioned previously, specific methods were employed for data processing in this context. After annotating and filtering the data, as described in Section 5.1, the masks were cropped and placed onto black images. No further modifications were made to the RGB images; the raw images were utilised as they were originally recorded.

Figure 17: Correctly prepared data for the CNN

With this data preparation, the following information was gathered. The diagrams below depict the distribution of frames throughout the entire dataset, divided into three categories:

- Train set: the dataset used to train the model
- Validation set ("Val set"): the dataset where the model's performance during training was measured.
- Test set: a fully separated set that the model has never encountered, the performance was assessed.

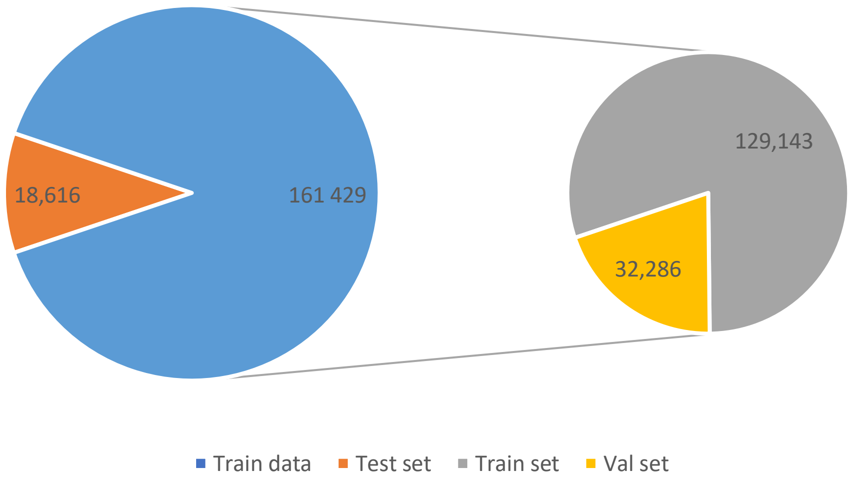

Figure 18: Distribution of frames

The dataset consists of images from 1,005 cattle, with the following breakdown:

- **Training dataset**: 905 cattle
- **Test dataset**: 100 cattle

Breakdown of the data:

- **Training data**:
  = **Train set** + **Validation set = Training data**
  = 129,143 images + 32,286 images **=** 161,429 images
- **All data**:
  = **Training data** + **Test data = all data**
  = 161,429 images + 18,616 images = 180,045 images.

5.2.1 Results

The results were measured on the **Test set**, where the **Mean Absolute Error (MAE)** was calculated for each individual animal and then averaged across all the cattle. The result was an MAE of **17.33**.

The following diagram illustrates the **predicted weight distribution** of the cattle.

Each animal in the dataset is linked to a specific number of frames, determined by the duration it was captured on camera. To standardise this, a sliding window method was used, selecting every consecutive set of five frames for each animal. For example, an animal with 100 frames would yield 20 data points, with each data point representing a unique five-frame segment. Probability density function (PDF) represents the likelihood of various weight values occurring in a population, indicating the distribution of the weights.

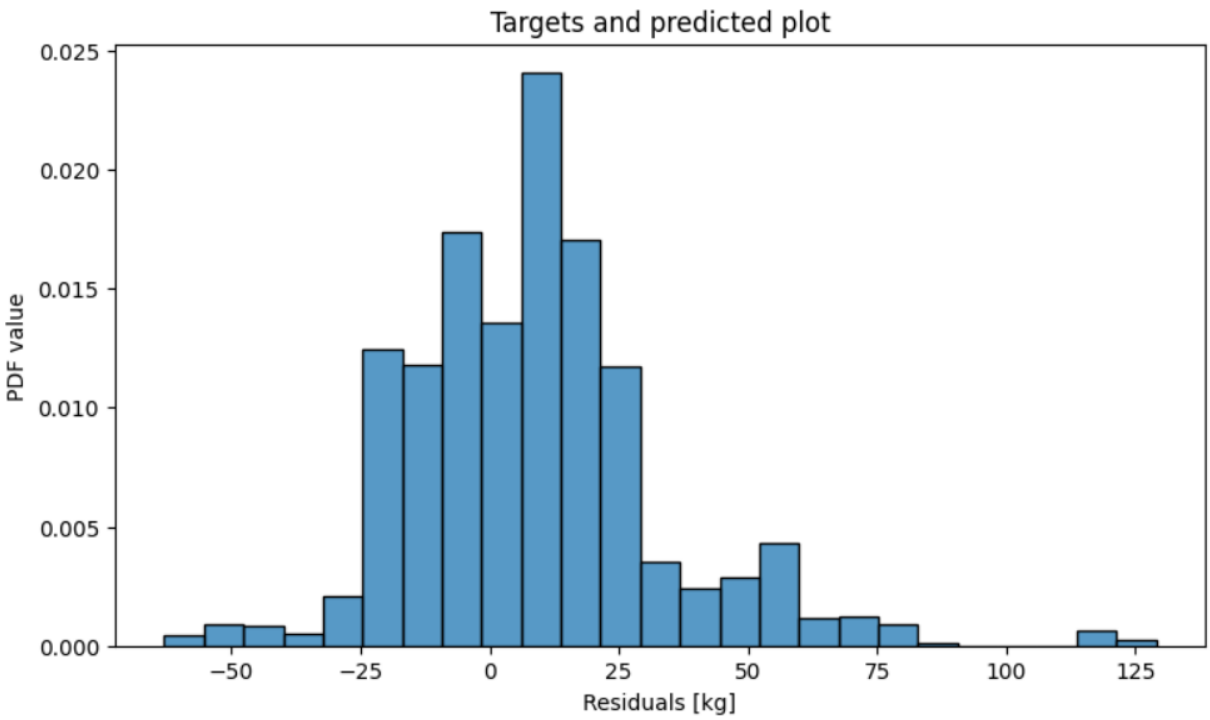

Figure 19: The predicted weight distribution of the cattle

The following table shows the percentage of cattle falling into each weight range based on the aforementioned “sliding window” evaluation.

	250- 300	300-350	350-400	400-450	450-500	500- 550	550- 600	600-650	650-700	700- 750	750-800
+-200kg	100	100	100	100	100	100	100	100			100
+-100kg	100	100	100	100	100	100	100	100			0
+-50kg	100	100	84.86	100	100	96.86	52.73	0			0
+-20kg	98.48	82.92	68.44	70.47	76.07	63	18.65	0			0
+-15kg	93.91	66.43	52.17	62.69	62.5	48.71	6.75	0			0
+-10kg	77.66	49.94	22.87	55.7	24.39	27.43	0.96	0			0
+-5kg	41.62	23.13	7.89	26.42	3.2	11.29	0	0			0
group size	197	843	621	386	656	700	311	20	0	0	26

Figure 20: The percentage of cattle in each weight range

The results indicate that the model slightly underestimates the weight of the cattle, particularly in animals with larger weights. This trend is primarily due to the lower representation of larger weight animals in the training dataset, as compared to the smaller weight animals, which results in less accurate predictions for the heavier cattle.

5.3 RGB cross-camera validation approach

With this approach, the CNN model was trained on the RGB data from the Dahua camera. Prior to training, however, modifications were made to the images. An algorithm was developed to minimise the fisheye distortion present in the RGB camera images, enabling inference and testing to be conducted on the undistorted images obtained from 3D cameras.

Figure 21: Left - Native image from the RGB camera, Right - Native image after the fish-eye distortion algorithm

This method helps to ensure that the RGB image closely resembles the 3D image in terms of dimensions as much as possible. After this, the same process was repeated: perform segmentation on the undistorted images using the previously trained neural network to obtain the masks, and then overlayed these masks onto black images.

In this case the data distribution is the following:

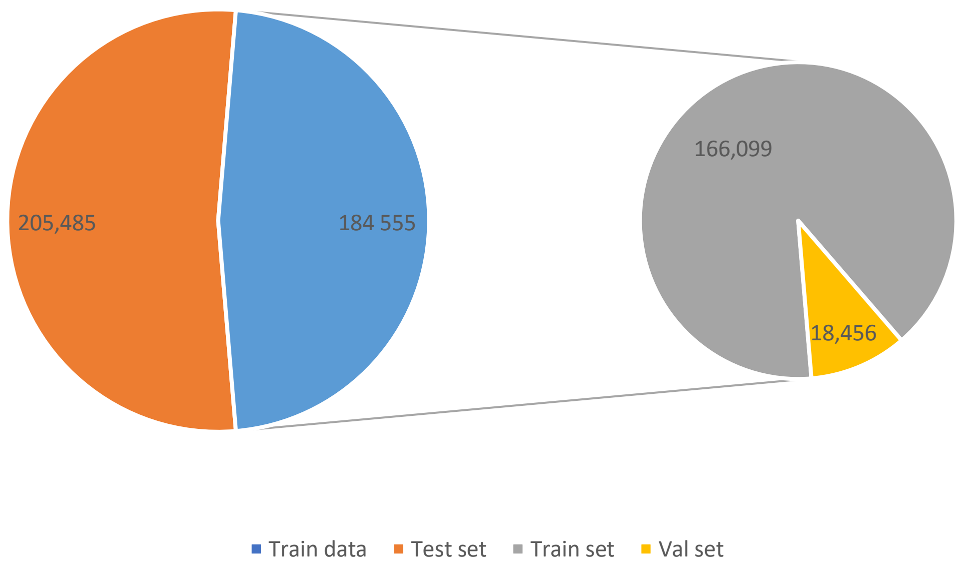

Figure 22: Distribution of frames

The training dataset consists of 973 cattle, while the test set comprises 813 animals. The total dataset consists of 390,040 images, broken down as follows:

- **All data** = 390,040
- **Train data** + **Test set** = 205,485 + 184,555
- **Train data** = **Train set** + **Val set** = 166,099 + 18,456

As noted earlier, the training set consists of data with minimised fisheye distortion, while the test set is based on data from the RGB channel of the 3D camera. Given that the cameras are different, their distortions also vary. To address this, the images from the 3D RGB channel were scaled up to match the dimensions of the masks derived from the RGB images, using an algorithm to ensure they were approximately the same size.

The two sets of images were captured almost simultaneously, making them nearly identical in size. However, because the colour sensors of the two cameras differ, the values of the various colour channels also differ. This discrepancy is evident in the two images and can affect the prediction accuracy of the neural network.

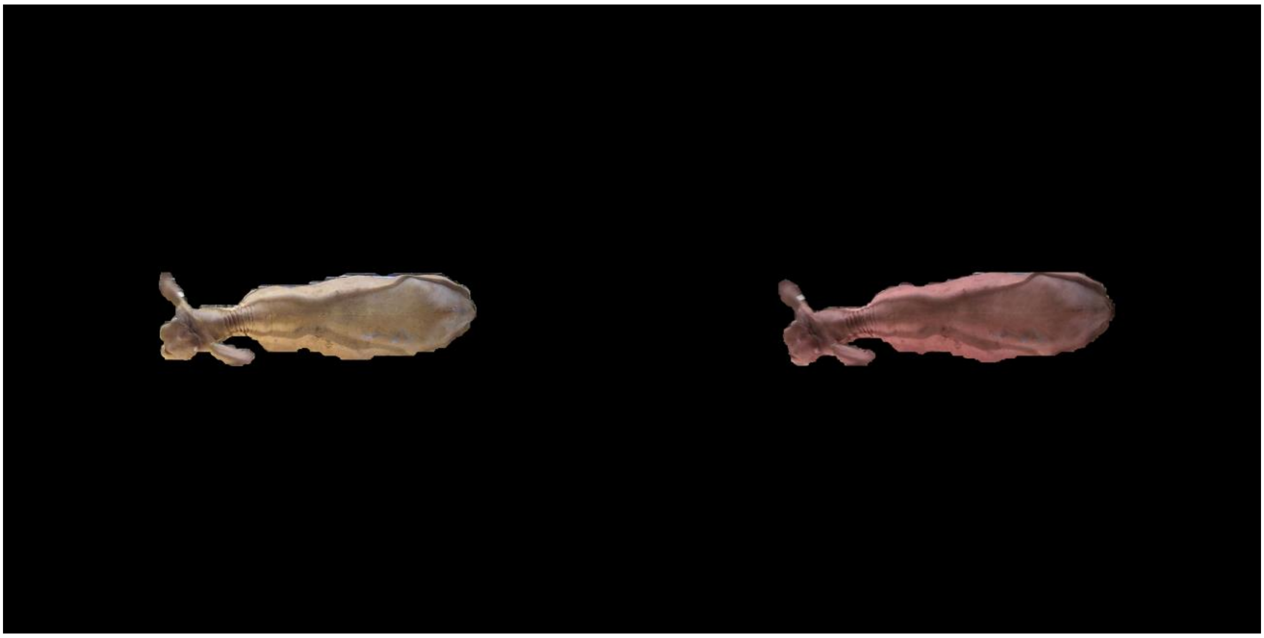

Figure 23: Left - RGB image with minimised fish-eye distortion, Right - 3D RGB image

Afterwards, inference was performed using the model trained on minimised fisheye RGB images on the scaled-up images from the 3D camera.

5.3.1 Results

The MAE was calculated for each individual animal, and then averaged across all of them to obtain the following result: **26.64**. The following diagram and table illustrate the predicted weight distribution of the cattle using the aforementioned sliding window approach.

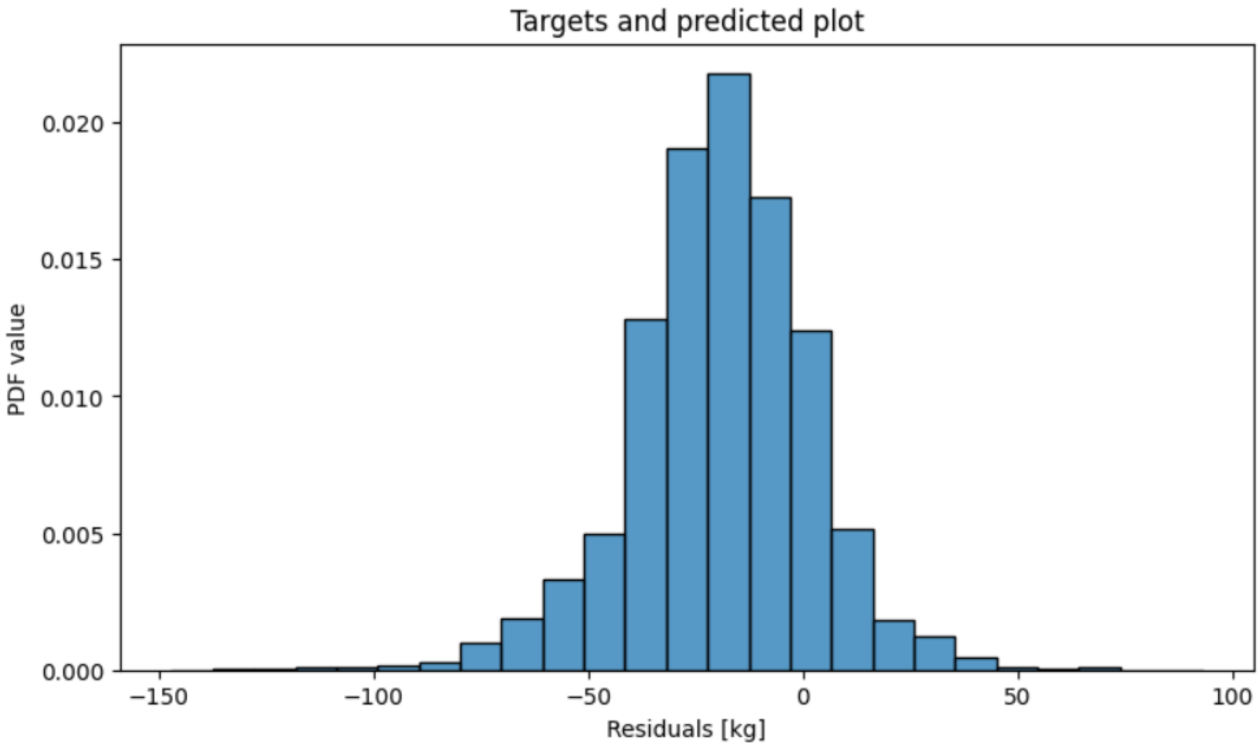

Figure 24: The predicted weight distribution of the cattle2

	[200, 250)	[250, 300)	[300, 350)	[350, 400)	[400, 450)	[450, 500)	[500, 550)
+-200kg	100	100	100	100	100	100	100
+-100kg	100	100	99.28	99.98	100	100	100
+-50kg	0	75.06	88.31	95.57	97.36	95.93	100
+-20kg	0	1.6	23.02	67.47	72.99	44.72	100
+-15kg	0	0.25	12.61	52.53	63.87	38.21	100
+-10kg	0	0	6.73	37.26	48.66	24.39	100
+-5kg	0	0	2.99	20.02	27.56	13.41	93.33
group_size	10	814	14719	19415	6208	246	15

Figure 25: The percentage of cattle in each weight range2

The model trained on data from entirely different cameras shows slightly reduced accuracy, yet it still achieves an average accuracy of 90% in weight estimation. The scaling, removal of distortion, and image transformation steps effectively mitigated the lens differences between the cameras. Therefore, the only plausible explanation for the slightly lower performance is the difference in colour adjustment. As shown in the image below, the same animal, captured at the same time by two different cameras, appears with different colour tones. This model tends to estimate darker animals as heavier than lighter animals, likely due to these colour discrepancies.

Figure 26: Left: RGB image with minimised fish-eye distortion, right: 3D RGB image

5.4 RGB cross camera validation – grayscale approach

To address the issue of differing colour adjustments between cameras, the RGB images were converted into grayscale. This conversion reduces the amount of information available to the neural network, but it can be advantageous in this case. By stripping away the colour information, the model is less influenced by the distortions and specific colour adjustments inherent to each camera lens, enabling it to generalise better across different camera systems.

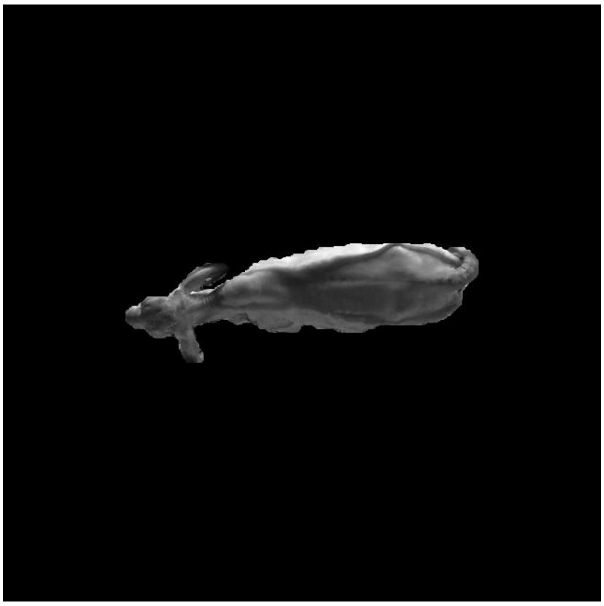

Figure 27: Grayscale image

5.4.1 Results

The MAE was calculated for each individual animal, and then averaged across all of them to obtain the following result: **16.73**. The following diagram and table illustrate the predicted weight distribution of the cattle using the sliding window approach as in previous cases.

	[0, 200)	[200, 250)	[250, 300)	[300, 350)	[350, 400)	[400, 450)	[450, 500)	[500, 550)
+-200kg		100	100	100	100	100	100	100
+-100kg		100	100	99.69	99.97	100	98.78	100
+-50kg		100	98.53	97.8	98.22	96.5	75.61	100
+-20kg		0	42.14	75.16	82.94	66.08	63.41	100
+-15kg		0	20.15	61.45	71.96	54.62	55.69	100
+-10kg		0	10.07	44.63	54.49	40.45	41.06	93.33
+-5kg		0	4.42	24.78	30.99	21.38	24.39	66.67
group_size	0	10	814	14719	19415	6208	246	15

Figure 28: The percentage of cattle in each weight range3

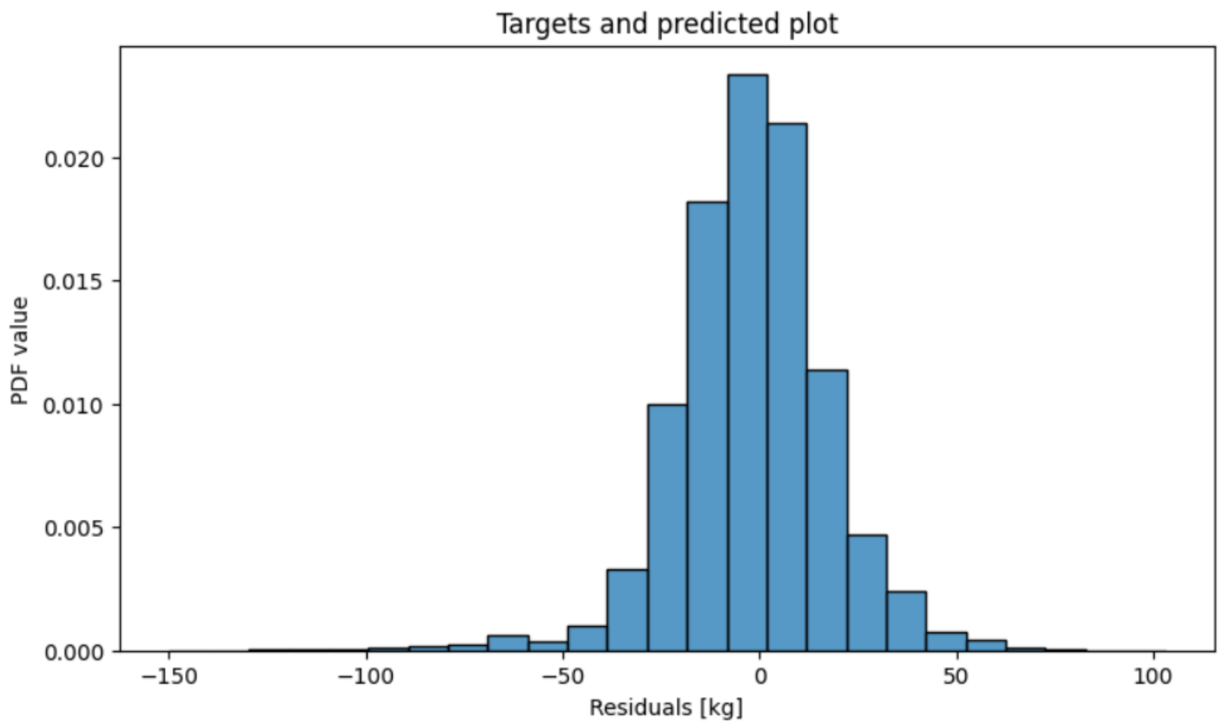

Figure 29: The predicted weight distribution of the cattle3

This approach successfully achieved the desired outcome of reducing weight overestimation. The Gaussian distribution of the results is closest to the ideal, with a mean value around zero and a very narrow shape. This indicates that the model's predictions align more closely with the parameters of the training data, providing the most realistic weight estimations.

5.5 Graphical interface

PigBrother developed a new AnimalSelector graphical user interface (GUI) designed to assist in animal selection. This GUI is adaptable for use with both pigs and cattle, with the only requirement being a change in the AI model behind it. The screens below show an instance of the GUI used for pig selection.

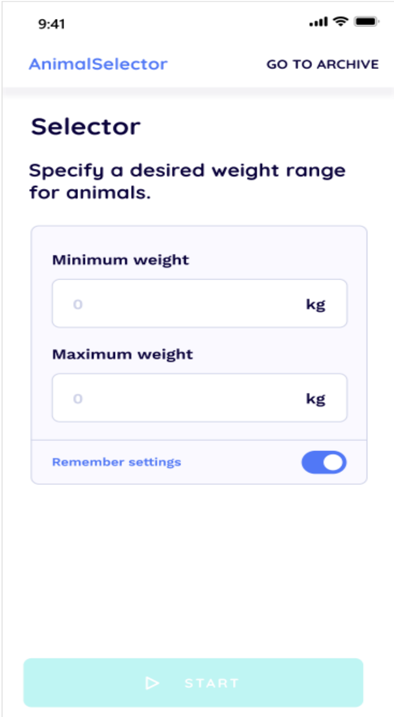

Figure 30: AnimalSelector - Screen 1

Screen 1 is the starting screen where the user can set the desired weight range for selection. The application will detect weights outside of this weight range, but they will be visualised as “underweight” and “overweight,” respectively.

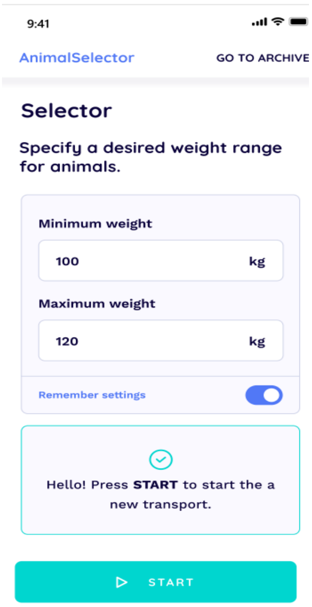

Figure 31: AnimalSelector - Screen 2

After setting the weights the user can start the camera and the animal weighing (screen 2).

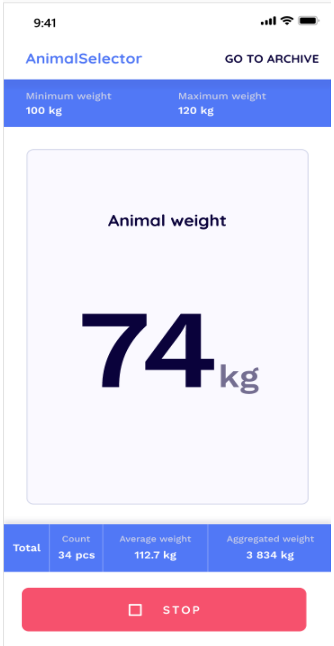

Figure 32: AnimalSelector - Screen 3

After weighing has started, the actual weight of the animal is visualised for the user (screen 3).

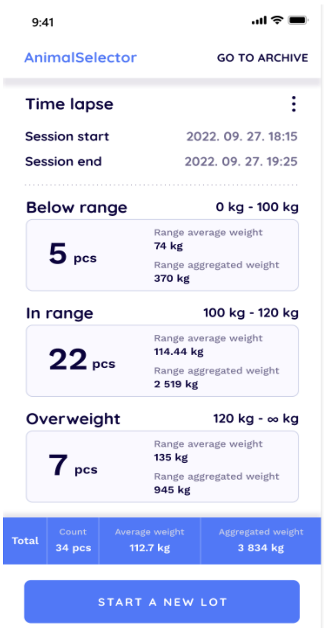

Figure 33: AnimalSelector - Screen 4

After clicking the STOP button, a summary page will appear (screen 4).

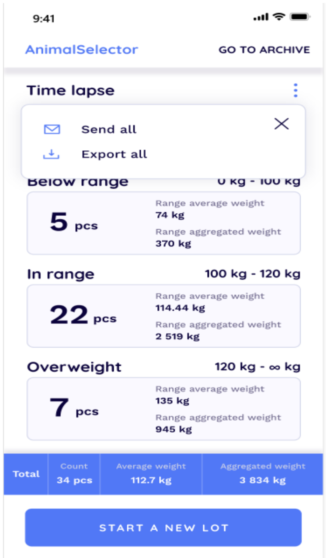

Figure 34: AnimalSelector - Screen 5

There is an Archive view where the users can look through the previous sessions and download an excel file containing the individual weights of these sessions (screen 6).

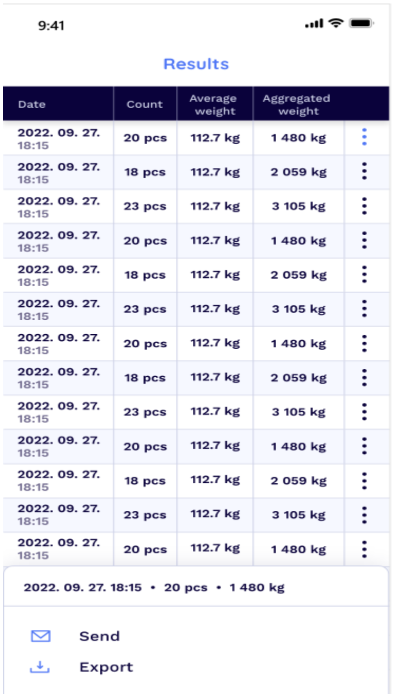

Figure 35: AnimalSelector - Screen 6

5.6 Use cases

The application is designed for specific use cases where end users require detailed information about the herd, such as weight distribution, average weight, total mass, or individual animal weights. These scenarios define how to deploy and test the solution in practical settings.

The CattleWeigherEstimator solution effectively addresses these needs by offering the following capabilities:

- Weighing individual animals with 90% accuracy.
- Creating annotated video containing timestamps, individual animal weights at each frame, and the associated truck number.
- Providing an easy-to-use and user-friendly graphical user interface (GUI), enabling quick and seamless control of the application.
- Allowing weighing to be performed on an edge server; once installed, no internet connection is required for operation.

5.7 Installation

The camera should be installed with a top-down view, positioned perpendicularly to the ground. After installation, PigBrother will perform camera calibration via the internet.

Although there is no specific camera required for the CattleWeigherEstimator solution, it must meet the following criteria:

- IP 67 rating (dustproof and waterproof).
- PoE (Power over Ethernet) for easier installation and power management.
- RTSP protocol support for streaming and integration.
- Low to moderate fish-eye effect to reduce distortion.
- Fixed focal length for consistent focus.
- Infrared capability to support low-light conditions.

While PigBrother can assist in selecting the best camera model from available options, it is recommended to identify suitable cameras in the market. If weighing of animals needs to occur during early morning or late evening hours, it is essential to collect training data at night as well to ensure accurate weight estimation under varying lighting conditions.

Figure 36: Camera positioning for cattle weight estimation

5.8 Maintenance

It is recommended to clean the camera lenses every two weeks or after extended periods of use. The application automatically manages the video data, deleting videos older than three weeks (this setting can be adjusted to meet the customer's needs).

5.9 How to improve accuracy

Collecting video data along with individual animal weights, integrated with RFID data, can further enhance the accuracy of the weight estimation model. This additional information allows for more precise calibration and adjustment of the model to reflect real-world conditions more accurately.

6. Statistical analysis of trained models

The collected data comprised actual and predicted weight measurements, along with timestamps, for more than 800 individual animals. The primary aim was to evaluate how the time of day and weight category influenced the accuracy of weight estimations using statistical analyses.

To ensure the data was clean, potential outliers were filtered using the Interquartile Range (IQR) method, applied to the absolute deviation values (the absolute difference between predicted and actual weight). For the purpose of statistical analysis, a relative accuracy measure was calculated as a continuous target variable.

The time of day was categorised based on the hour of the day, and the weight category was defined by creating equal 100 kg bins ranging from 200 kg to 500 kg. The collected data contained actual and predicted weight figures as well as timestamps for more than 800 distinct animals. It was aimed at assessing the effect of time of day and weight category using statistical methods. Data were first filtered from potential outliers by applying the IQR method on the values of absolute deviation (absolute value of the difference between predicted and true weight). For statistical purposes, a relative accuracy measure was calculated as a continuous target variable. The data was then grouped into these mixed categories. For descriptive purposes, a boxplot was created, which shows the mean and range of the data points in each case (group).

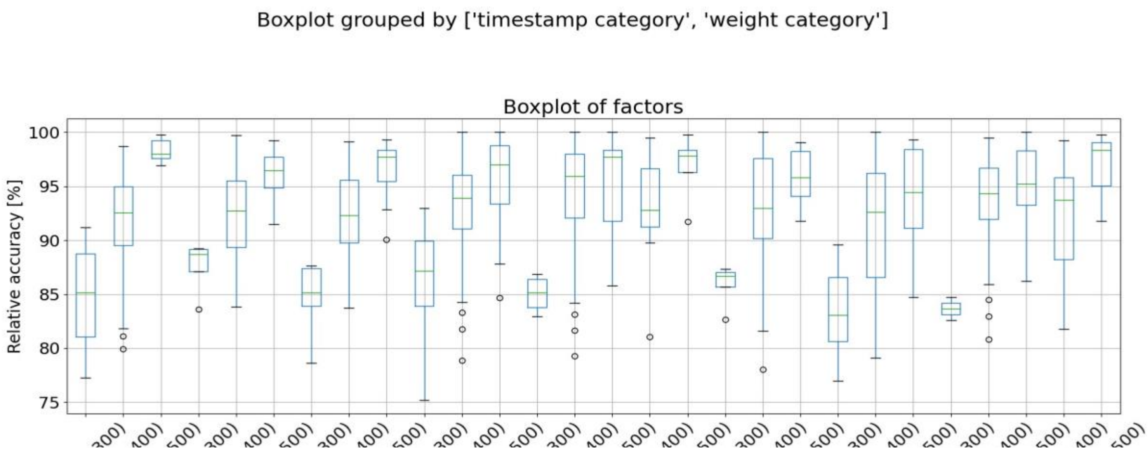

Figure 37: Boxplot of categorical variables

The contingency table below shows the animal count among the groups.

timestamp category [h]	200-300 kg	300-400 kg	400-500 kg
7	4	55	7
8	4	54	8
9	8	104	18
10	7	144	30
11	4	69	25
12	0	7	4
13	5	67	13
14	5	60	12
15	2	41	11
16	0	22	3

Figure 38: Animal counts in groups

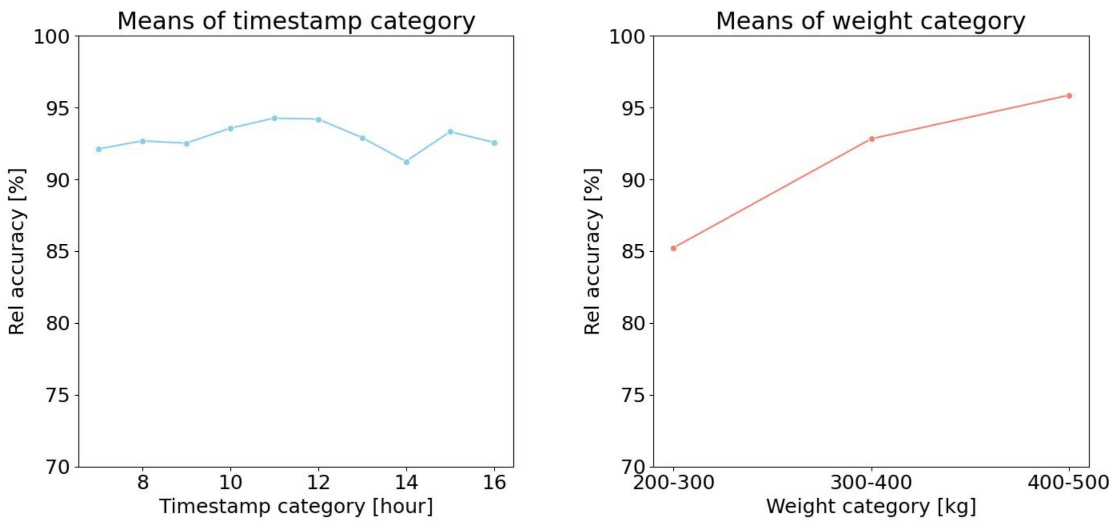

Figure 39: Marginal means

The marginal means of timestamp categories and weight categories were then calculated across the groups. Fig. 39 shows a clear positive trend for the weight category. It means that the accuracy of the weight estimation strongly depends upon the weight of the animal. According to the analysis, animals with weight between 300 and 500 kg are estimated with accuracy higher than 90%.

The timestamp category shows approximately constant behaviour with accuracy continuously higher than 90%. It means that there is no significant influence of the time of day when the image was created upon the accuracy of the weight estimation.

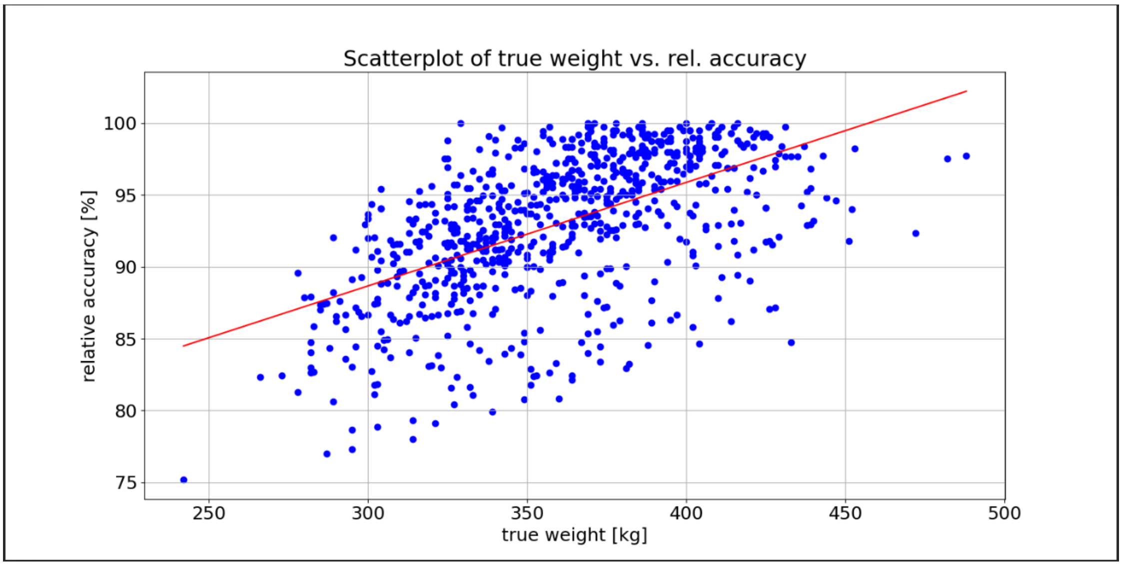

Figure 40: Correlation plot

The Pearson correlation of true weights versus the relative accuracy was calculated. The correlation coefficient was moderate and significant (correlation coefficient: 0.5586, p-value=2.8478e-66). Thus, the Pearson correlation analysis confirms the previous result, that the trained model performs better for larger weights. For more detailed analysis the ANOVA (Analysis of Variance) method was performed for the two categories considering only the main effects. The table contains the following: sum of squares for model terms (sum_sq), degrees of freedom (df), F statistic value, and p-values.

	sum_sq	df	F	PR(>F)
C(timestamp_category)	374.338657	9.0	2.108102	2.66e-02
C(weight_category)	3262.061737	2.0	82.666903	2.722e-33
Residual	15409.251660	781.0		

Figure 41: ANOVA table

The analysis showed that both the weight category and the timestamp category can be considered significant at the 95% confidence level. However, the weight category plays a larger role in determining the accuracy or performance of the model because it explains much more of the variability in the data. This is reflected in its high F-statistic and sum of squares, which indicate that weight has a stronger influence on model performance. Even at a 98% confidence level, the weight category remains highly significant, underscoring its importance. In contrast, while the timestamp category is still significant, it explains less variance in the model's performance.

7. Next steps

7.1 Further Broome site testing

To conduct further testing at the Broome site, two servers need to be connected to the internet to allow PigBrother to deploy the latest version of the software. Local operators would be required to reinstall the counting camera in a top-down position on the race leading to the truck. Ideally, testing should be conducted in a time zone where PigBrother can provide remote monitoring and assistance. User acceptance tests would also need to be scheduled and conducted. Once the user acceptance tests were successfully completed, the application should be ready for cattle counting and weighing. In case any tests fail, PigBrother would devise a plan to address the issues promptly.

7.2 Other site tests

PigBrother is willing to engage with businesses who are interested in trialling the technology to discuss the opportunities.

7.3 Improving accuracy

Enhancing the accuracy of the weight estimation model would need to leverage a video collection system integrated with individual weighing and RFID data. To further refine the application:

- Collect more data under varying lighting conditions.
- Gather data across different weight ranges.
- Ensure individual weights are obtained using a cattle crush, ideally with second-precision extraction of weight and RFID information from this machine.

The following steps are necessary to advance the application:

1. Install a server at the partner’s site.
2. Position cameras in a top-down orientation.
3. Connect the server and cameras to the same local LAN network.
4. Grant remote access to PigBrother for software installation and configuration.
5. Allow PigBrother to remotely calibrate the cameras.

To improve overall accuracy, a 2-4 hour-long video from the site with animals can be used to retrain the model to understand the site’s environmental conditions. If issues with accuracy arise during the application's use, the customer can contact PigBrother and provide footage of the errors for model retraining. Additionally, if there is frequent transportation at night, these specific lighting conditions should be addressed separately.

7.4 Adoption and commercialisation

PigBrother will reach out to businesses that express interest in purchasing and installing AI-powered cattle counting and weighing technologies trialled in this proof-of-concept project. PigBrother will schedule an online meeting to discuss opportunities, challenges, hardware requirements, and deployment strategies.

Following this discussion, businesses can specify their preferred applications and the number of cameras they wish to install. PigBrother will then create a detailed schedule outlining the hardware installation process, software release timelines, user acceptance test schedules, and associated deadlines to ensure clarity and coordination.