Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/issn/15375110

Research Paper

# Image-based body mass prediction of heifers using deep neural networks

Roel Dohmen [a], Cagatay Catal [b,*], Qingzhi Liu [a]

[a] Wageningen University & Research, Information Technology Group, Wageningen, the Netherlands
[b] Qatar University, Department of Computer Science & Engineering, Doha, Qatar

ARTICLE INFO

*Article history:*
Received 2 August 2020
Received in revised form
30 January 2021
Accepted 3 February 2021
Published online 19 February 2021

*Keywords:*
Deep learning
Computer vision
Body weight prediction
Convolutional neural network

Manual weighing of heifers is time-consuming, labour-intensive, expensive, and can be dangerous and risky for both humans and animals because it requires the animal to be stationary. To overcome this problem, automated approaches have been developed using computer vision techniques. In this research, the aim was to design a novel mass prediction model using deep learning algorithms for youngstock on dairy farms. The Mask-RCNN segmentation algorithm was used to segment the images of heifers and remove the background. A convolutional neural networks (CNN) model was developed on the Keras platform to predict the body mass of heifers. For the case study, a new dataset based on images of 63 heifers was built. Animals were between the age of 0 and 365 days and lived on the same farm in the Netherlands. The range of body mass of the heifers was between 37 kg and 370 kg. The side-view model had a coefficient of determination ($R^2$) of 0.91 and a Root Mean Squared Error (RMSE) of 27 kg, the top-view model had an $R^2$ of 0.96 and an RMSE of 20 kg. The experimental results demonstrated that our proposed mass prediction model using the Mask-RCNN segmentation algorithm, together with a novel CNN-based model, provided remarkable results, and that the top view was more suitable than the side view for predicting the body mass of youngstock in dairy farms.



## 1. Introduction

To calculate the body mass of animals, morphological traits such as chest girth, withers height, and hip height can be measured by farmers, or weighing using electronic scales can be applied. The latter method is more accurate because it determines the true mass by placing the animal on a scale but the former that uses several relations between certain body dimensions, such as waist girth and withers height can be more convenient (Heinrichs & Losinger, 1998). However, this type of measurement is still time-consuming and labour-intensive; thus automated approaches based on machine learning and computer vision-based approaches have been developed. Although these estimates are less precise compared to the weighing using scales, they provide less labour-intensive ways of estimating the body mass of the animal. The body mass can also be estimated from 3D images based on animal volume (Le Cozler et al., 2019).

The precision of weighing systems can vary with errors for animals varying from 1 to 10 kg. Also, in ruminants, the body

* Corresponding author.
E-mail addresses: roel.dohmen@wur.nl (R. Dohmen), ccatal@qu.edu.qa (C. Catal), qingzhi.liu@wur.nl (Q. Liu).
https://doi.org/10.1016/j.biosystemseng.2021.02.001
1537-5110/

**Nomenclature**

| Abbreviation | Description |
|---|---|
| API | Application programming interface |
| CNN | Convolutional neural networks |
| DL | Deep learning |
| FCN | Fully connected network |
| fps | Frames per second |
| GAN | Generative adversarial networks |
| GBT | Gradient boosting tree |
| KNN | K-Nearest neighbour |
| LSTM | Long-short term memory |
| MAPE | Mean absolute percentage error |
| MLP | Multi-Layer perceptron |
| MT | Model tree |
| R−CNN | Region-based CNN |
| RF | Random forest |
| RMSE | Root mean square error |
| ROI | Region-of-interest |
| SMO | Sequential minimal optimisation |
| SVR | Support vector regression |

mass can vary considerably within the day. In dairy heifers, to estimate real body mass, measurements are usually carried out twice a day and the average value used. Body mass is mainly used to determine feed allowance and used with regards to expecting growth.

To make smart decisions, the estimation of animal body mass is a very beneficial tool because the current mass of a heifer can impact several management decisions. For instance, the mass can be used to determine whether the heifer is healthy or whether its heifer is within the expected margins. This kind of mass prediction can be used to interpret if the animal is on track with respect to its growth requirements because the heifer should reach an optimal body mass for starting lactation after 24 months.

Recently, deep learning, a sub-branch of machine learning, has provided remarkable results in different complex tasks such as face recognition and object detection; however, these techniques are less widely used in precision agriculture and they have not been investigated for mass estimation of heifers. The objective of our study is to explore the potential of deep learning approaches for estimating the body mass of dairy heifers. Another important objective is to design the system using low-cost devices (i.e., the cost of the device should be less than 100 €) instead of expensive cameras, and therefore, 2D images were used in this study. This cost criterion was set during our initial discussions with the owner of the dairy farm used in this research. Thus, our research was "can we estimate the body mass of heifers by 2D images using deep learning algorithms?"

To the best of our knowledge, no study has applied the Mask R−CNN algorithm together with a CNN-based model for body mass prediction of heifers. Our study is therefore different than the previous studies in the literature that used other techniques (Alonso et al. (2013); Huma and Iqbal (2019); Miller et al. (2019); Shahinfar et al. (2020); Huang et al. (2019)). Also, this new dataset is larger than most of the datasets applied in the literature.

The contributions of this study are.

- A novel body mass prediction model for heifers using deep learning algorithms
- A new body mass prediction dataset for heifers
- Better performance for building deep learning-based body mass prediction models for heifers

## 2. Related work

Different techniques have been proposed and validated for animal mass prediction based on 2D and 3D vision-based techniques in the literature. While there exist some machine learning-based studies, the number of deep learning-based approaches in this domain is still quite limited (Dohmen et al., 2021). In a recent systematic literature review (SLR) study (Dohmen et al., 2021), 26 papers that applied computer vision techniques for body mass estimation of livestock were reviewed. Seven features, namely top view body area, withers height, hip height, body length, hip-width, body volume, and chest girth were widely used in these approaches.

In this section, some of the studies that have applied machine learning algorithms and deep learning algorithms are presented.

Alonso et al. (2013) used the support vector regression (SVR) algorithm for predicting the carcass mass of a beef breed from the North of Spain (i.e., Asturiana de los Valles breed cattle) and report that their model can predict carcass weights 150 days before the slaughter day. Huma and Iqbal (2019) predicted the body mass of the Balochi sheep breed of Pakistan using machine learning techniques, but they did not use deep learning algorithms. Miller et al. (2019) used artificial neural networks (ANN) algorithms to predict live mass and carcass characteristics of beef cattle and showed that 3D imaging coupled with the ANN algorithm can predict the live body mass and carcass characteristics of live animals. Shahinfar et al. (2020) evaluated multi-layer perceptron (MLP), model tree (MT), random forest (RF), and support vector machines (SVM) with sequential minimal optimisation (SMO) for predicting the carcass traits of Korean Hanwoo beef cattle. They showed that SVM with SMO provides relatively better performance. However, their focus was not to calculate the actual weight of the cattle.

There are several studies that applied machine learning algorithms in predicting the body mass of different animals such as sheep, chickens, rabbits (Ali et al., 2015; Mortensen et al., 2016; Salawu et al., 2014; Szyndler-Nędza et al., 2016).

Huang et al. (2019) applied deep learning and transfer learning approaches for body dimension measurements of Qinchuan cattle; however, their focus was not to estimate the live body weight of the cattle, and they did not use CNN algorithms. They used Kd-network that is a deep learning architecture designed for 3D model recognition tasks. Shahinfar et al. (2019) applied deep learning (DL), gradient boosting tree (GBT), K-nearest neighbour (KNN), model tree (MT), and random forest (RF) algorithms to predict sheep carcass traits

from early-life records, but the aim was not to predict the live body mass of sheep, and they did not give any additional information about their deep learning model. They showed that the RF algorithm provides the best performance, among others. Cang et al. (2019) propose a deep learning model based on the Faster R–CNN algorithm for body mass estimation and show that their approach can estimate masses accurately. However, they did not focus on heifers and did not apply the Mask R–CNN algorithm for object segmentation. Jensen et al. (2018) applied CNN algorithms for estimation of live body mass and reported an $R^2$ of 0.95. They collected data only from 17 animals, which is one of the limitations of their study. They did not focus on heifers and did not apply the state-of-the-art Mask R–CNN segmentation algorithm. Qiao et al. (2019) proposed a new instance segmentation approach based on the Mask R–CNN algorithm for precise cattle instance segmentation. They also did not focus on mass prediction, but their approach can be used as part of the mass prediction model.

## 3. Methodology

### 3.1. Convolutional Neural Networks (CNN)

CNN were designed for object recognition tasks such as face recognition. Krizhevsky et al. (2012) designed a CNN model that provided state-of-the-art results on the image classification task. The following benefits of CNN models exist compared to a fully connected neural network model (Brownlee, 2016, 2019):

- Fewer parameters had to be learned.
- They are invariant to distortion and object position in the image.
- They learn features automatically.

CNN is a neural network model specifically proposed for 2D image data but can be applied on 1D and 3D data as well. The innovation of CNN models is that they can learn multiple features in parallel. There are typically three types of layers in CNN models, which are listed as follows (Brownlee, 2016, 2019):

- *Convolutional layers:* These layers consist of filters and feature maps. Filters are considered as the neurons of this layer, and filters create an output value depending on the weighted inputs. A feature map is considered as the output of one layer, which is applied to the previous layer. Each movement of the filter is an activation of the neuron, and this type of output is stored in the feature map.
- *Pooling layers:* These layers are used to down-sample the feature map of previous layers and generalise feature representations. These kinds of layers help to reduce the overfitting by applying simple techniques such as taking the average of the input value. Pooling layers mostly follow one or more convolution layers to consolidate the previously learned features.
- *Fully-connected layers:* These layers are the layers used in typical feedforward neural networks. They are used at the end of a CNN model for making predictions. Whilst the convolution layer addresses feature extraction and the pooling layer consolidates these features, the fully-connected layer is responsible for making predictions.

The AlexNet model that was developed based on the CNN algorithm won the ImageNet challenge in 2012, and this model, which consists of 8 layers, rose the interest in CNN algorithms (Ballester & Araujo, 2016). Later, more complex models such as ResNet that includes 152 layers were developed (Wu et al., 2019). Recently different open-source software platforms and libraries were developed, such as Keras, TensorFlow, PyTorch, Caffe, Theano, MXNET, CNTK, and DeepLearning4J (Nguyen et al., 2019). Keras and TensorFlow are the most used software libraries in different application domains. Keras is a high-level neural network application programming interface (API) that supports several deep learning engines such as TensorFlow and Theano (Gulli & Pal, 2017). In this research, we developed our CNN models using the Keras platform because it is easy to build models, user friendly, has different production deployment options, supports multiple GPUs, and is integrated with the TensorFlow deep learning engine. Different model configurations were implemented and investigated to reach optimal configuration settings.

Apart from the CNN-based algorithms, there are other kinds of deep learning algorithms applied in different application domains. Recurrent neural networks (i.e., long-short term memory (LSTM)), generative adversarial networks (GAN), autoencoders, deep belief networks, and restricted Boltzmann machines are some of the well-known other deep learning algorithms. However, CNN is the most used one amongst these algorithms.

### 3.2. Region-based CNN models

In the book of Brownlee (2019) concerning deep learning for computer vision, the author explained how deep learning algorithms could be used for several challenging computer vision tasks. According to Brownlee (2019), object detection is a challenging task that aims to identify the presence, location, and type of objects in an image. This complex problem includes several subproblems, namely object recognition, object localisation, and object classification. Object recognition addresses where the objects are, object localisation finds their extent, and object classification specifies what they are. There is an extension of object detection, which is called object segmentation. In object segmentation, pixels that belong to each detected object are marked. This problem is different from using bounding boxes during object localisation. Compared to object detection, object segmentation is considered to be a more difficult problem.

For the object detection problem, deep learning approaches have recently achieved remarkable results. The region-based convolutional neural network (R–CNN) is a family of CNN-based algorithms designed for object detection. The following four variations of this R–CNN algorithm have been developed: R–CNN (Girshick et al., 2014), Fast R–CNN (Girshick, (2015)), Faster R–CNN (Ren et al., 2016), and Mask R–CNN (He et al., 2017).

In the R–CNN algorithm, the selective search algorithm is applied to suggest bounding boxes, features are discovered with the CNN, and object classifications are performed with the linear support vector machines algorithm. In the Fast R–CNN algorithm, a region-of-interest (RoI) pooling layer is applied after the CNN algorithm, and both class labels and RoI are predicted. In Faster R–CNN, a region proposal network evaluates features discovered from the CNN and learns to suggest RoI. Mask R–CNN, which is an extension of Faster R–CNN, uses an additional output model for predicting the mask per object. The Mask R–CNN is the most recent of these algorithms and supports both object detection and object segmentation. Whilst R–CNN models are more accurate compared to other models, they can be slow for real-time prediction. For instance, the YOLO (Redmon et al., 2016) algorithm is faster, but less accurate compared to R–CNN models. Depending on the problem, the object detection algorithm must be selected.

Mask R–CNN algorithm consists out of two parts: Faster R–CNN for object detection and classification and a fully connected network (FCN) for semantic segmentation. Faster R–CNN is a bounding box object detection approach that creates regions-of-interest (RoI) using a region proposal network and applies pooling to each RoI in an image to determine the class of the object within that RoI. After the pooling, the RoI with the object label is presented. Mask R–CNN, however, does not use the pooling approach because it can cause alignment problems for the image segmentation. As such, mask R–CNN uses an align layer for feature extraction instead of a pooling layer. Parallel to object detection and classification, an FCN is used to determine which pixels belong to the object within the RoI. This FCN creates a binary mask that aligns with the RoI.

There can be three use cases of the Mask R–CNN application (Brownlee, J. (2019)):

1. *The use of a pre-trained model:* In this approach, a pre-trained model that was trained on a large set of images is used for a new image dataset.
2. *The generation of a new model via transfer learning:* Again, a pre-trained model is used, but it is customised for a new dataset using transfer learning approaches.
3. *The development of a new model from scratch:* A new model is developed from scratch for a new dataset.

In this research, the first use case was followed and a pre-trained model that was trained on the Common Objects in Context (COCO) dataset was used. The dataset can be accessed from the following link: https://cocodataset.org.

A pre-trained model has model weights that are loaded before making predictions.

### 3.3. *Our approach*

The methodology of our research is presented in Fig. 1. Since there is no public dataset available that consists of heifer images with their corresponding weights, a new dataset had to build as part of our study. Our case study was performed on a conventional dairy farm in the south of the Netherlands that had 150 mature dairy cows and 63 heifers between the age of 0 and 365 days. The images of animals were taken from two angles, one is from the top view, and the other one is from the side view of the heifer. The inputs for our prediction models were 2D binary images with a resolution of 640 × 480 pixels.

Twenty-four different models were trained and optimised for each view. These models were then validated by using a separate validation set, and $R^2$, mean absolute percentage error (MAPE), and root mean square error (RMSE) values were calculated for each model. The best model for each angle was used, and the performance of a top-view based model and a side-view based model were compared.

The Mask-RCNN algorithm, which is used for object detection (He et al., 2017), was utilised. With the help of the Mask-RCNN algorithm, the images of heifers were segmented and removed from the background. Later, a CNN body mass prediction model was used to estimate the mass of heifers. Different kinds of layers, such as the convolution layer, pooling layer, and fully connected layer, were investigated whilst building the CNN-based model. The best performing model was selected for final implementation.

Data were collected on a farm in the south of the Netherlands. This farm was preferred for the experiments because the first author had contacts on this farm, accessibility to the farm was relatively easy, and there was also a wide variation in the body mass of animals aging from 1 month to 1 year of age. For this study, 63 heifers of the cross bread of Holstein Friesian, Montbéliarde, and Swedish red between the age of 0–365 days were selected. As such, heifers were a three-way rotational cross bred and partly Holstein, Montbéliarde, and Swedish red. The range of body masses of the heifers was between 37 kg and 370 kg. The histogram of the body masses with a bin size of 10 kg is presented in Appendix B. Identification numbers and heritage information cannot be provided in detail because the animals had to be anonymised due to the privacy policy of the company. All the actions performed with the animals followed the standard calm handling methods used to handle the cattle on the farm and, as such, did not result in any stress for the animals.

The animals were weighed using a scale; afterwards, they were recorded from two viewpoints using a webcam. The experimental setup is shown in Fig. 2. In the picture, the heights of both cameras are indicated, and the distance of the side-view camera to the centre of the walking path of the animal is shown. The distance of the top view camera from the ground was 3100 mm. The distance of the side view camera to the ground was 910 mm, and the distance to the middle of the walking alley is 2300 mm.

The first step was to weigh the animals with a scale using an accuracy of 1 kg. It was calibrated using several metal blocks that had a combined known weight of 50 kg. The animals were immobilised in a box with closed sidewalls and semi-transparent gates. After the weighing, the gate at the front was opened, and animals were guided past two Microsoft HD-3000 LifeCam webcams. These cameras were placed in such a way that they took a video from the side and top. The video resulting from this process had a resolution of 96 dpi, a dimension of 640 × 480, and a recording speed of 30 fps (frames per second).

To build the deep learning-based body mass prediction model, images had to be extracted from the corresponding

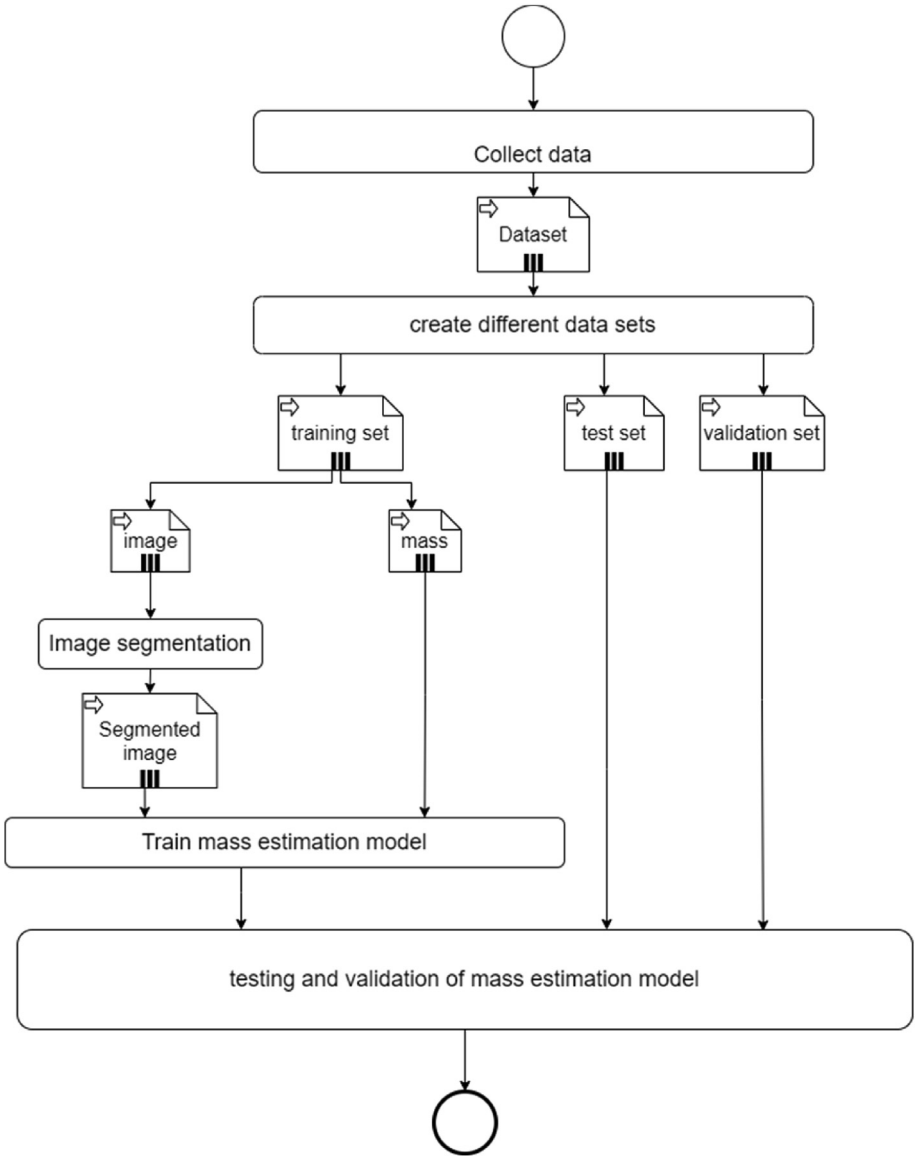



**Fig. 1 – Overview of the processes of the case study.**

video. A Python script, which imports the video files and extracts the individual frames from the video, was implemented, and later, images were saved. From these images, the images where the animal was completely in the frame were selected and used to create the dataset. This resulted in two datasets, one with the side view images and one with top view images. Experimental measurements (images and body mass recording) were performed once per animal and all the data were recorded on the same day because similar conditions for all the animals and pictures should have been satisfied.

As shown in Fig. 3, the extracted images were processed using the Mask-RCNN algorithm, which segments the images to extract a mask that contains a form of the animal and creates a binary image from that mask. The mask-RCNN model was first trained using the COCO dataset to develop a model that can segment animals in images. Since some segmentation results from the mask-RCNN model had quality problems, and did not segment the animal accurately, a manual selection procedure was also performed to select the best images for each animal. For every animal, five images were segmented, from these images, one image was selected where the legs and the head of the animal were adequately visible. Images of two animals were removed from the dataset due to the fact that the images were poor-quality and did not represent the animal silhouette clearly, for example, no legs or head visible. Because the data collection and data processing stages were performed at different times, animals that were shown in poor-quality images were dismissed. It would require extra time and effort if we wished to include these two heifers at a later stage as well, however, our assumption was that the remaining set would be still sufficient to represent the heifers on the farm.

The binary images and their body masses were linked based on the experiment number of each animal. The final dataset that consists of binary images and corresponding masses was then split into the following sets: 60% for the

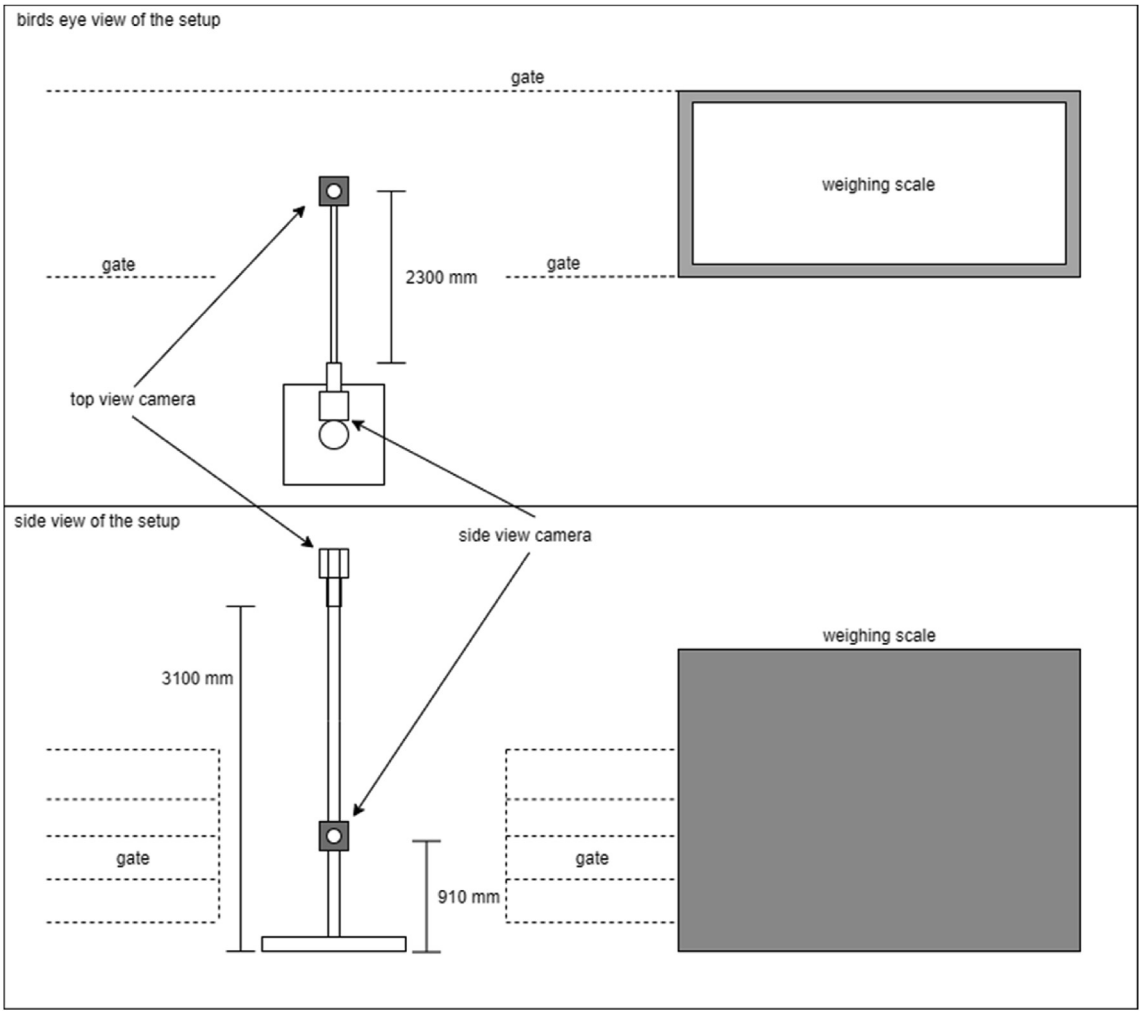



**Fig. 2 – Multi-viewpoint sketch of the data collection setup.**

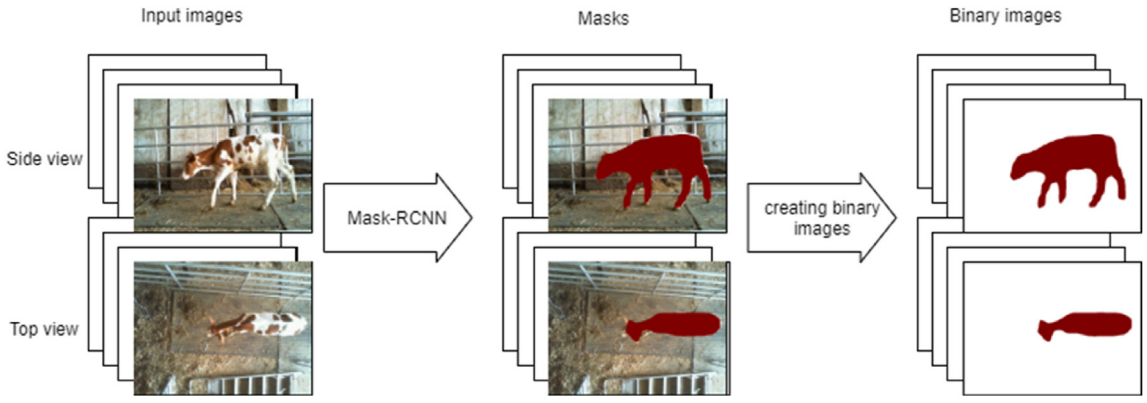



**Fig. 3 – The image preparation process used to create the binary images.**

training set, 20% for the testing set, and 20% for the validation set (Alay & Al-Baity, 2020).

Python 3.6 and Keras deep learning library version 2.3.1 on top of TensorFlow 2.0 were used for the implementation of the models. 48 individual models were designed (i.e., 24 for the side-view mass prediction and 24 for the top-view weight prediction). The basic design of the CNN used is presented in Fig. 4. The mass prediction models were built using several combinations of convolution layers combined with max-pooling layers. These combinations were then followed by a flattening layer that converted the data from the last convolution operation in a one-dimensional array to use in the dense layer that performs the regression for calculating the body mass of the heifer in the image.

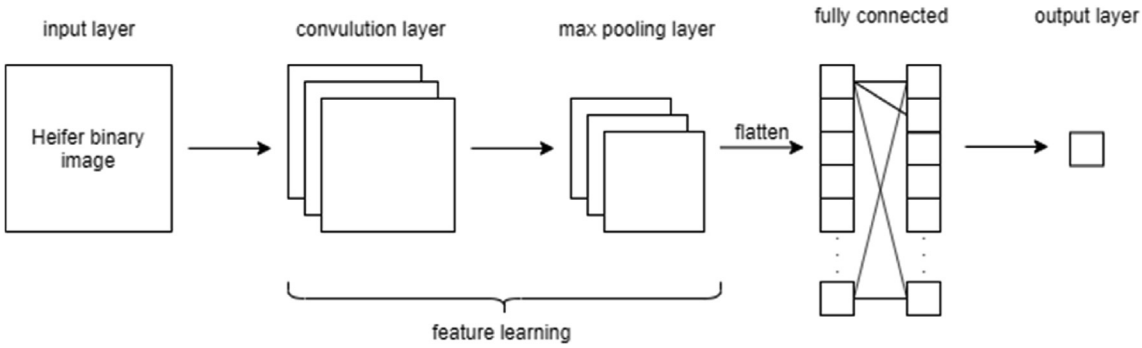



**Fig. 4 – Design of the convolutional neural networks.**

The models were trained by using the training set to find the model parameters. For optimising the performance, the ADAM optimiser (Kingma & Lei Ba, 2015) was used, and the MSE was selected as the loss function. The MSE was then calculated by comparing the prediction value and the real body mass of the heifer. The models were trained and tested for 30 iterations, which was determined after plotting the loss function against the number of iterations. It showed that the loss function converged after 20 epochs. In addition, to make sure that the optimal model is reached, the number of iterations was increased by 50%. During the training of each model, the best model configuration was saved for the validation of models and to find out which models provided the best performance for the mass prediction of heifers.

To find out which model configurations worked best, two parameters were changed in the training process. The first parameter is the number of convolution and max-pooling layers. The second parameter is the number of filters used per convolution layer.

There were four main model designs; the first model started with one convolution layer and one max pooling layer. For every following model, there was one feature learning part consisting of one convolution layer and one max pooling layer added. The second parameter was based on $2^n$, where n was the filter number. Six filter counts were applied during this research, which means that the first model had six filters for the convolution layer that were calculated by filling the number of the filters in for n in $2^n$, which resulted in six models with one convolution layer. The number of filters used in each layer is shown in Table 1.

The next step was to evaluate the performance of the individual models. This was performed by using a separate validation set. The model configurations that were created during the training process were loaded and used to predict the body mass of the heifers in the dataset. The real mass and the predicted weight were then exported to an excel file, and the $R^2$ (Eq. (1)), RMSE (Eq. (2)), and MAPE (Eq. (3)) values were calculated. These values were assessed, and based on the RMSE and MAPE, the best model for the weight prediction was selected.

$$R^2 = 1 - \frac{\sum\left(W_i^{predicted} - W_i^{real}\right)^2}{\sum\left(W_i^{real}\right)^2} \quad (1)$$

$$RMSE = \sqrt{\frac{\sum\left(W_i^{predicted} - W_i^{real}\right)^2}{N}} \quad (2)$$

$$MAPE = \sum\left|\frac{W_i^{predicted} - W_i^{real}}{W_i^{real}}\right| \times \frac{100}{N} \quad (3)$$

**Table 1 – Calculated number of filters displayed for every filter number.**

| Filter number | Layer 1 | Layer 2 | Layer 3 | Layer 4 |
|---|---|---|---|---|
| 1 | 2 | 4 | 8 | 16 |
| 2 | 4 | 8 | 16 | 32 |
| 3 | 8 | 16 | 32 | 64 |
| 4 | 16 | 32 | 64 | 128 |
| 5 | 32 | 64 | 128 | 256 |
| 6 | 64 | 128 | 256 | 512 |

## 4. Experimental results

### 4.1. Model configurations for top-view images

Table 2 shows the performance of the different model configurations based on the four evaluation parameters, namely $R^2$, MAE, MAPE, and RMSE. It is shown that based on the $R^2$ model, 1.3.5 has the worst fit for the validation dataset, and this model does not fit well with the validation data. The best model in this table is model 1.4.3 based on $R^2$, MAPE, and RMSE.

### 4.2. Model configurations for side-view images

Table 3 presents the performance of the different model configurations based on the four evaluation parameters, namely $R^2$, MAPE, and RMSE. It was shown that model 2.4.1 has the worst performance, and the best one is model 2.3.5. Furthermore, it is clear that the performance of weight prediction using the CNN algorithm is high based on the overall high $R^2$ and low error rates.

### 4.3. Accuracy of prediction models

When Tables 2 and 3 were compared, it can be seen that the $R^2$ value was better for model 1 than in model 2. This happened because of the large errors in the side-view pictures of animals. In Fig. 5, the measured/predicted masses of

**Table 2 – Performance of the top-view model configurations. The colour coding goes from green (best value) to red (worst value).**

| Model number. | R2 | MAPE | RMSE |
|---|---|---|---|
| 1.1.1 | 0.94 | 0.13 | 25.30 |
| 1.1.2 | 0.96 | 0.12 | 21.07 |
| 1.1.3 | 0.96 | 0.11 | 21.20 |
| 1.1.4 | 0.97 | 0.13 | 20.47 |
| 1.1.5 | 0.96 | 0.13 | 21.55 |
| 1.1.6 | 0.94 | 0.13 | 24.51 |
| 1.2.1 | 0.94 | 0.13 | 24.47 |
| 1.2.2 | 0.96 | 0.13 | 21.81 |
| 1.2.3 | 0.96 | 0.11 | 20.97 |
| 1.2.4 | 0.95 | 0.11 | 22.21 |
| 1.2.5 | 0.97 | 0.11 | 21.54 |
| 1.2.6 | 0.97 | 0.14 | 20.52 |
| 1.3.1 | 0.93 | 0.13 | 24.78 |
| 1.3.2 | 0.95 | 0.12 | 21.17 |
| 1.3.3 | 0.96 | 0.12 | 21.54 |
| 1.3.4 | 0.94 | 0.12 | 23.01 |
| 1.3.5 | **0.93** | **0.18** | **27.65** |
| 1.3.6 | 0.96 | 0.12 | 21.60 |
| 1.4.1 | 0.95 | 0.15 | 24.60 |
| 1.4.2 | 0.94 | 0.13 | 24.24 |
| 1.4.3 | **0.96** | **0.11** | **19.57** |
| 1.4.4 | 0.94 | 0.15 | 24.99 |
| 1.4.5 | 0.96 | 0.12 | 20.38 |
| 1.4.6 | 0.94 | 0.15 | 24.92 |
| Best values | 0.97 | 0.11 | 0.94 |

**Table 3 – Performance of the side-view model configurations.**

| Model number. | R2 | MAPE | RMSE |
|---|---|---|---|
| 2.1.1 | 0.83 | 0.15 | 33.22 |
| 2.1.2 | 0.86 | 0.12 | 29.61 |
| 2.1.3 | 0.88 | 0.11 | 28.37 |
| 2.1.4 | 0.88 | 0.10 | 29.48 |
| 2.1.5 | 0.88 | 0.11 | 28.08 |
| 2.1.6 | 0.88 | 0.12 | 27.87 |
| 2.2.1 | 0.85 | 0.15 | 30.88 |
| 2.2.2 | 0.86 | 0.13 | 30.13 |
| 2.2.3 | 0.86 | 0.12 | 29.55 |
| 2.2.4 | 0.88 | 0.11 | 27.64 |
| 2.2.5 | 0.88 | 0.11 | 27.63 |
| 2.2.6 | 0.89 | 0.11 | 28.99 |
| 2.3.1 | 0.86 | 0.13 | 30.48 |
| 2.3.2 | 0.86 | 0.13 | 30.09 |
| 2.3.3 | 0.86 | 0.11 | 30.68 |
| 2.3.4 | 0.86 | 0.10 | 30.62 |
| 2.3.5 | **0.91** | **0.10** | **26.68** |
| 2.3.6 | 0.86 | 0.13 | 30.00 |
| 2.4.1 | **0.77** | **0.17** | **38.78** |
| 2.4.2 | 0.85 | 0.12 | 30.71 |
| 2.4.3 | 0.87 | 0.13 | 28.71 |
| 2.4.4 | 0.88 | 0.13 | 28.46 |
| 2.4.5 | 0.85 | 0.13 | 30.74 |
| 2.4.6 | 0.85 | 0.14 | 31.54 |
| Best value | 0.91 | 0.10 | 26.68 |

the animals are presented. Each value belongs to a single animal. When assessing the performance of the models, it is observed that the top-view model performs better with respect to the $R^2$ parameter, with a value of 0.96 compared to model 2.3.5 with a value of 0.91. With respect to the RMSE parameter, model 2.3.5 performs worse than model 1.4.3 (i.e., 26.68 kg vs. 19.57 kg). As such, model 1.4.3 is a better model for predicting the body mass of heifers until the age of 365 days.

## 5. Discussion

The results show that the top view was better for predicting the body mass of the animal when using a CNN-based mass prediction model. This difference might be related to the variation in segmentation quality of the side view images. In previous studies, a good prediction result was also achieved from the top-view (Kashiha et al., 2014). In some of the side view images, it was observed that the legs and head were not segmented accurately. In the top view images, this is not a problem since the legs of the animals were not in the picture; as such, they did not need to be segmented. This sensitivity due to the bad segmentation error might be solved by eliminating the legs in the images and only taking into account the trunk of the animal (Nishide et al., 2018) (Kashiha et al., 2014). Because legs were not present, an estimation of body mass with or without legs should be performed in the future and the correlation between the results obtained from the two approaches should be calculated.

In this study, the mask R–CNN algorithm was used for image segmentation. However, for image segmentation, the histogram of oriented gradients (HOG) method could be used as features and support vector machines (SVM) used as the classifier. The main advantage of the mask R–CNN algorithm is that it can perform detection, classification, and image segmentation (Xu et al., 2020), and therefore, different applications such as welfare monitoring can be implemented easily. The Mask R–CNN image segmentation approach can

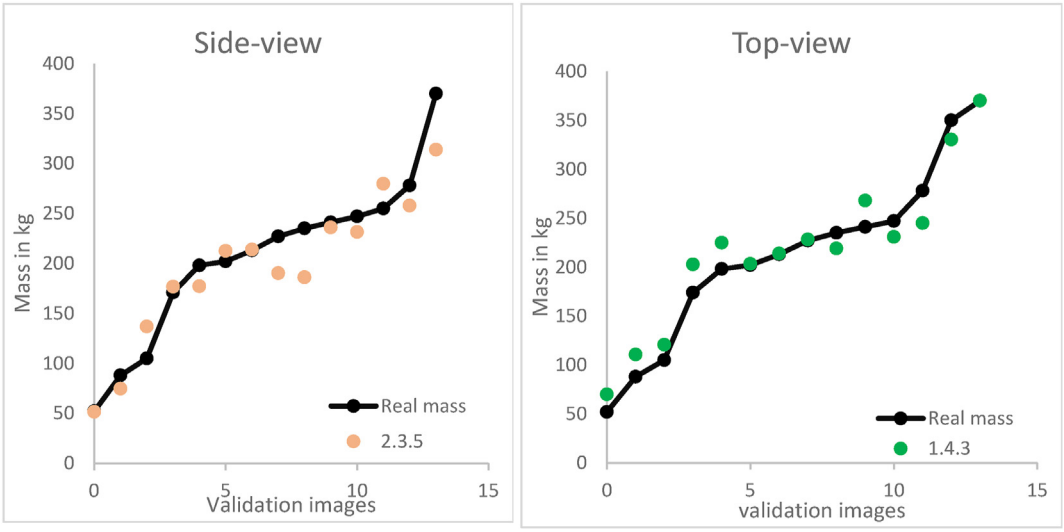



**Fig. 5 – Two plots of the values for best fitting models built based on the side-view and top-view. The x-axis displays the index of the animal and the y-axis the measured/predicted mass of the animal.**

also be applied for different applications such as lameness detection. Since our envisioned scenario was to develop on-farm welfare monitoring systems, the mask R–CNN algorithm was used for image segmentation and the CNN algorithm for mass prediction.

The data for this research was collected on a single farm, and therefore different datasets may be needed for different cow breeds. Additional datasets must be built based on multiple dairy farms. Since there is a high variation in housing systems and farming procedures (van der Peet et al., 2018), one farm cannot be a representative for variations that occur between the herds of different farms and this could be a limitation. The second limitation is that the model was designed based on the animals from a few specific crossbreeds of dairy cows. This means that the model may not perform as expected when applied to the other breeds. The interest in crossbreds is also to introduce large variations, that could be of interest when considering applying the approach to other breeds. In these animals, a small error percentage causes too many variations in mass (e.g., from 50 to 400 kg, a 5% error means variations in kg of 2.5–20 kg). Different cow breeds can differ significantly in their body composition, and the features discovered here might not be relevant for different breeds. To improve the variation captured in the prediction model, it would be beneficial to gather data from different dairy farms in the Netherlands.

A third limitation is that the dataset does not have an equal distribution of body masses (i.e., 37 kg–370 kg). As shown in 0, between 110 and 170 kg, there are significantly fewer data points. This can cause poor mass estimations for this range. It might be beneficial to gather data on animals that are in the mass range of 110 kg and 170 kg. The fourth limitation is the fact that the model is trained on a dataset that consists of images that are acquired from a fixed distance to the animal. A processing step needs to be added to scale the pictures.

The system must also be tested on older heifers (i.e., after insemination) and also dairy cows, since body mass could be of interest to better manage dairy herds.

## 6. Conclusions and future work

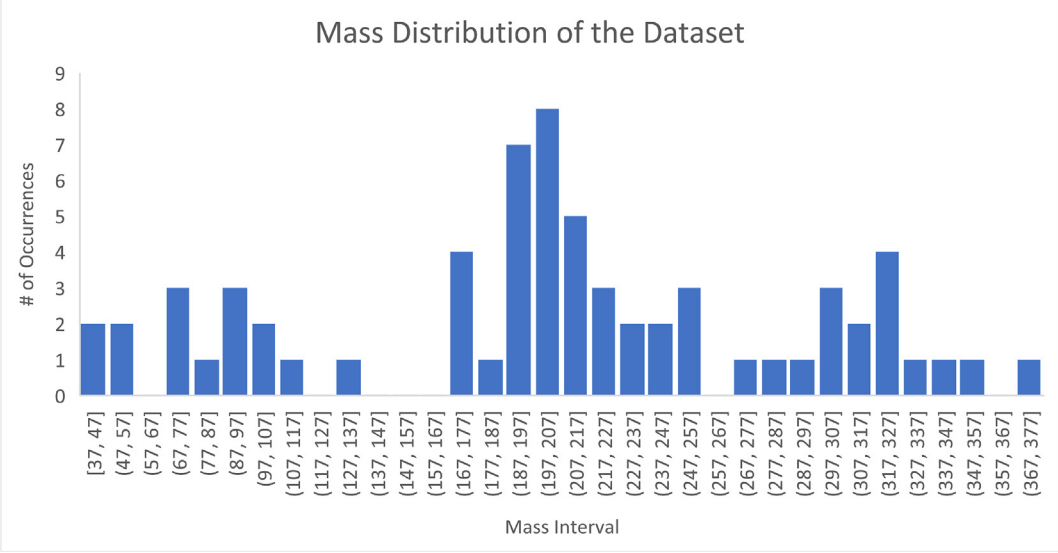
Our experimental results on the validation set achieved an $R^2$ value of 0.96 and an RMSE of 20 kg. Compared to the previous studies reported in the literature (O Ozkaya & Bozkurt, 2009), our model is therefore promising. It can be concluded that the combination of the mask R–CNN algorithm with CNN-based prediction algorithms is an effective approach for predicting the body mass of heifers. Also, it was demonstrated that the top view images provides better performance compared to the side view images, and high-performance prediction models can be built with 2D images.

For future work, our prediction model can be analysed on different datasets in different farms to create a prediction model that is more generic for the variation of heifers. The selection of these farms should be performed carefully to cover all the farming systems. The variation between breeds and within breeds needs to be accounted for, combined with the main diet that the animals get on the different farms. To increase the performance of models, more image processing techniques can be investigated to improve performance.

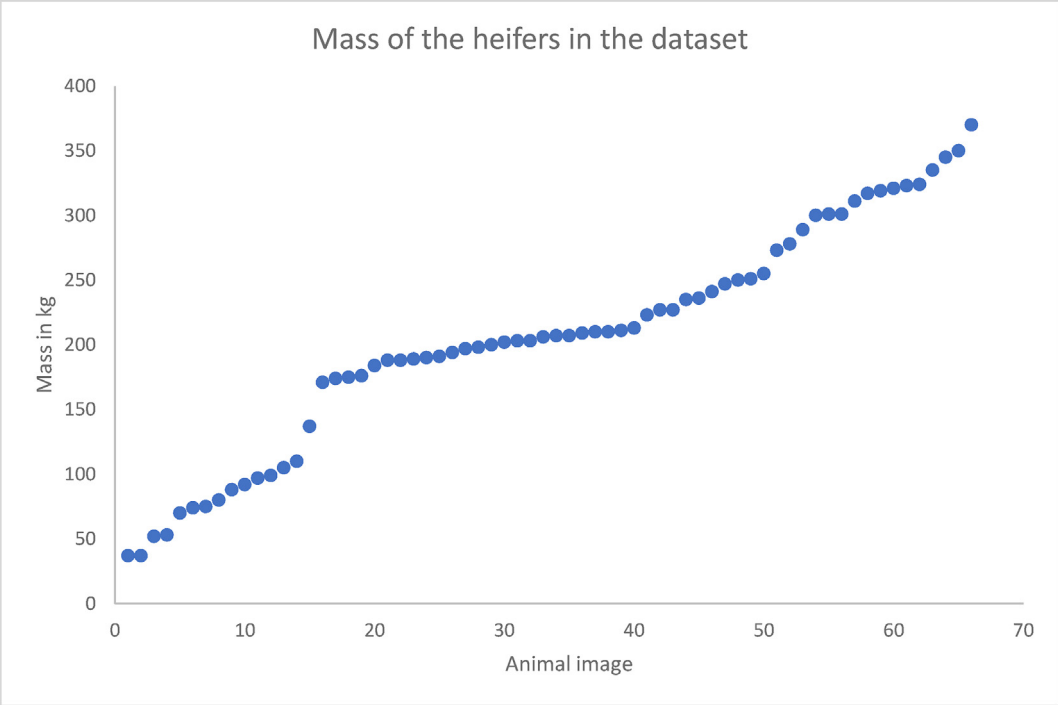
## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Appendix A. Mass distribution of the dataset




## Appendix B. Histogram of the mass with a bin size of 10 kg



## REFERENCES


Alay, N., & Al-Baity, H. H. (2020). Deep learning approach for multimodal biometric recognition system based on fusion of Iris, face, and finger vein traits. *Sensors, 20*(19), 5523.

Ali, M., Eyduran, E., Tariq, M. M., Tirink, C., Abbas, F., Bajwa, M. A., & Shah, S. H. (2015). Comparison of artificial neural network and decision tree algorithms used for predicting live weight at post weaning period from some biometrical characteristics in Harnai sheep. *Pakistan Journal of Zoology, 47*(6).

Alonso, J., Castañón, Á. R., & Bahamonde, A. (2013). Support Vector Regression to predict carcass weight in beef cattle in advance of the slaughter. *Computers and Electronics in Agriculture, 91*, 116–120.

Ballester, P., & Araujo, R. (2016, February). On the performance of GoogLeNet and AlexNet applied to sketches. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 30). No. 1.

Brownlee, J. (2016). *Deep learning with Python: develop deep learning models on Theano and TensorFlow using Keras*. Vermont, Australia: Machine Learning Mastery.

Brownlee, J. (2019). *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Vermont, Australia: Machine Learning Mastery.

Cang, Y., He, H., & Qiao, Y. (2019). An intelligent pig weights estimate method based on deep learning in sow stall environments. *IEEE Access, 7*, 164867–164875.

Dohmen, R., Catal, C., & Liu, Q. (2021). Computer vision-based weight estimation of livestock: A systematic literature review. *New Zealand Journal of Agricultural Research*, 1–21. Published online on 20 January 2021. https://www.tandfonline.com/doi/full/10.1080/00288233.2021.1876107

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440–1448).

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).

Gulli, A., & Pal, S. (2017). *Deep learning with Keras*. Packt Publishing Ltd.

He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961–2969).

Heinrichs, A. J., & Losinger, W. C. (1998). Growth of Holstein dairy heifers in the United States. *Journal of Animal Science, 76*(5), 1254–1260. https://doi.org/10.2527/1998.7651254x

Huang, L., Guo, H., Rao, Q., Hou, Z., Li, S., Qiu, S., & Wang, H. (2019). Body dimension measurements of qinchuan cattle with transfer learning from LiDAR sensing. *Sensors, 19*(22), 5046.

Huma, Z. E., & Iqbal, F. (2019). Predicting the body weight of Balochi sheep using a machine learning approach. *Turkish Journal of Veterinary and Animal Sciences, 43*(4), 500–506.

Jensen, D., Dominiak, K., & Pedersen, L. (2018). Automatic estimation of slaughter pig live weight using convolutional neural networks. In *Proc. 2nd int. Conf. Agro BigData decis. Support syst. Agricult* (pp. 1–4).

Kashiha, M., Bahr, C., Ott, S., Moons, C. P. H., Niewold, T. A., Ödberg, F. O., & Berckmans, D. (2014). Automatic weight estimation of individual pigs using image analysis. *Computers and Electronics in Agriculture, 107*, 38–44. https://doi.org/10.1016/j.compag.2014.06.003

Kingma, D. P., & Lei Ba, J. (2015). Adam: A method for stochastic optimization. *ICLR*, 4, 2015.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems, 25*, 1097–1105.

Le Cozler, Y., Allain, C., Xavier, C., Depuille, L., Caillot, A., Delouard, J. M., & Faverdin, P. (2019). Volume and surface area of Holstein dairy cows calculated from complete 3D shapes acquired using a high-precision scanning system: Interest for body weight estimation. *Computers and Electronics in Agriculture, 165*, 104977.

Miller, G., Hyslop, J., Barclay, D., Edwards, A., Thomson, W., & Duthie, C. A. (2019). Using 3D imaging and machine learning to predict liveweight and carcass characteristics of live finishing beef cattle. *Frontiers in Sustainable Food Systems, 3*, 30.

Mortensen, A. K., Lisouski, P., & Ahrendt, P. (2016). Weight prediction of broiler chickens using 3D computer vision. *Computers and Electronics in Agriculture, 123*, 319–326.

Nguyen, G., Dlugolinsky, S., Bobák, M., Tran, V., García, Á. L., Heredia, I., & Hluchý, L. (2019). Machine learning and deep learning frameworks and libraries for large-scale data mining: A survey. *Artificial Intelligence Review, 52*(1), 77–124.

Nishide, R., Yamashita, A., Takaki, Y., Ohta, C., Oyama, K., & Ohkawa, T. (2018). Calf robust weight estimation using 3D contiguous cylindrical model and directional orientation from stereo images. In *ACM international conference proceeding series* (pp. 208–215). Association for Computing Machinery. https://doi.org/10.1145/3287921.3287923.

O Ozkaya, S., & Bozkurt, Y. (2009). The accuracy of prediction of body weight from body measurements in beef cattle. *Archives of Animal Breeding, 52*(4), 371–377.

Qiao, Y., Truman, M., & Sukkarieh, S. (2019). Cattle segmentation and contour extraction based on Mask R-CNN for precision livestock farming. *Computers and Electronics in Agriculture, 165*, 104958.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779–788).

Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(6), 1137–1149.

Salawu, E. O., Abdulraheem, M., Shoyombo, A., Adepeju, A., Davies, S., Akinsola, O., & Nwagu, B. (2014). Using artificial neural network to predict body weights of rabbits. *Open Journal of Animal Sciences, 4*, 182–186.

Shahinfar, S., Al-Mamun, H. A., Park, B., Kim, S., & Gondro, C. (2020). Prediction of marbling score and carcass traits in Korean Hanwoo beef cattle using machine learning methods and synthetic minority oversampling technique. *Meat Science, 161*, 107997.

Shahinfar, S., Kelman, K., & Kahn, L. (2019). Prediction of sheep carcass traits from early-life records using machine learning. *Computers and Electronics in Agriculture, 156*, 159–177.

Szyndler-Nędza, M., Eckert, R., Blicharski, T., Tyra, M., & Prokowski, A. (2016). Prediction of carcass meat percentage in young pigs using linear regression models and artificial neural networks. *Annals of Animal Science, 16*(1), 275–286.

van der Peet, G., Leenstra, F., Vermeij, I., Bondt, N., Puister, L., & van Os, J. (2018). *Feiten en cijfers over de Nederlandse veehouderijsectoren 2018 (No. 1134)*. Wageningen Livestock Research.

Wu, Z., Shen, C., & Van Den Hengel, A. (2019). Wider or deeper: Revisiting the resnet model for visual recognition. *Pattern Recognition, 90*, 119–133.

Xu, B., Wang, W., Falzon, G., Kwan, P., Guo, L., Sun, Z., & Li, C. (2020). Livestock classification and counting in quadcopter aerial images using Mask R-CNN. *International Journal of Remote Sensing, 41*(21), 8121–8142.