IET Journals
The Institution of Engineering and Technology

A CNN-based Cow Interaction Watchdog

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

H. Ardö[1] *O. Guzhva*[2] *M. Nilsson*[1] *A. H. Herlin*[2]

[1] *Centre for Mathematical Sciences, Lund University, Sölvegatan 18, Lund, Sweden*
[2] *Swedish University of Agricultural Sciences, Department of Biosystems and Technology, Box 103, 23053 Alnarp, Sweden*
** E-mail: ardo@maths.lth.se*

Abstract: In the field of applied animal behaviour, video recordings of a scene of interest are often made and then evaluated by experts. This evaluation is based on different criteria (number of animals present, an occurrence of certain interactions, the proximity between animals and so forth) and aims to filter out video sequences that contain irrelevant information. However, such task requires a tremendous amount of time and resources, making manual approach ineffective. To reduce the amount of time the experts spend on watching the uninteresting video, this paper introduces an automated watchdog system that can discard some of the recorded video material based on user-defined criteria. A pilot study on cows was made where a Convolutional Neural Network (CNN) detector was used to detect and count the number of cows in the scene as well as include distances and interactions between cows as filtering criteria. This approach removed 38% (50% for additional filter parameters) of the recordings while only losing 1% (4%) of the potentially interesting video frames.

1 Introduction

Scientists working with animal behaviour and welfare are interested in studying the social interactions between cows in dairy farms. Typically these studies are performed by defining a set of interactions such as head butting, body pushing, social licking, and other physical and non-physical contacts. Every interaction is described in detail in a protocol, so called ethogram. Then an expert studies the area of interest for a considerable amount of time and counts the number of each interaction occurring or the duration of the interaction. This is performed for the whole duration of the video sequence/sequences used for the particular study [11]. Some of these interactions are relatively rare, which means that much expert time has to be spent on looking at raw video data to find potentially interesting sequences.

Some recent studies on cow behaviour in the dairy barn environment were based on different GPS or wireless sensor network (WSN) solutions [13]. These solutions allow scientists to see spatial distribution of animals and to measure varying levels of activity [13]. However, the number and complexity of behaviours that could be monitored with the position-based approach is limited. Therefore, new methods based on video surveillance and image analysis, which could extend the number of parameters for studying, are of great importance. Correct classification of interactions is of importance as some interactions are positive, friendly and strengthen the bond between animals while others are possibly deleterious and can reflect a struggle for resources.

In this paper, the objective was to take the first step towards an automated system for behavioural analysis. The study area was filmed using video cameras. Then an automated watchdog system removed unnecessary parts (e.g. those, containing events that were not relevant or without animals in the scene) of the recorded video material. The remaining video sequences still have to be studied by experts, but the time spent looking at uninteresting video sequences could be significantly reduced.

This was achieved by developing a cow detector that detects cows in the scene and represents them with rotated bounding boxes. That means that the number of cows could be counted and distances between the cows could be measured. Rotated bounding boxes allow much more precise distance measures between diagonally oriented cows as compared to more classical non-rotated bounding boxes due to the elongated aspect ratio of the cows.

The pilot study used to develop this watchdog was performed in a dairy barn in the south of Sweden with 252 Swedish Holstein cows. Cows were milked by four automatic milking stations (AMS), which had a common waiting area (6x18 meters). This waiting area is a common space which cows that are ready for milking are allowed to enter at any point in time. They may then interact with each other in order to decide the order of entry to each of the milking robots. A direct relationship between cows' inter-cow distance and their aggressive behavioural patterns was demonstrated by [2]. Other studies [3, 7, 9] showed certain effects of inferior animal welfare connected to the restrained performance of their natural behaviour. Therefore, early diagnostics of unconditional changes in animal behaviour when linked to health and welfare could not only save time and money for the farmer, but also decrease the production pressure for every animal in the barn [16].

1.1 Related work

Most previous work on detecting cows by using video cameras have been focused on monitoring areas where the orientation of the cows was known due to physical limitations imposed by the surroundings. Two examples of this are the Viola-Jones based detector of Arcidiacono et al. [21] for detecting cows at the feed barrier and the work of Martinez-Ortiz et al. [12] to detect and track cow heads in narrow entrance corridors. There are also approaches that rely on more advanced sensors, such as Abdul et. al [1] that uses a depth camera.

The current state of the art for detecting cows freely moving around was presented by Porto et al. [17]. They use a Viola-Jones based detector and need 6 cameras at 4.6 meters height to cover a 15.4×3.8 m area and detects cows in 3 different orientations: vertical, horizontal and diagonal with a hit rate of 90%. They do not use separate detectors for the two different diagonals which means that producing rotated bounding box from their results would not be straight forward. In the work presented in this paper, only 3 cameras at 3.6 meters height were needed to cover a 18×6 m area and detect cows in 32 different orientations with a hit rate of 97%. Note that the hit rates are from different datasets, and that the dataset used in this article have larger variations in viewpoints due to the use of fewer cameras at a lower height to cover a larger area.

General purpose object detection frameworks such as YOLO [18, 19] and SSD [10] have very good performance. They do, however, focus on detecting objects of varying size and aspect ratio but with fixed orientation. In the scenario considered in this paper, the size and aspect ratio are fixed and known, while orientation (rotation) varies and have to be estimated.

He et al. [6] have considered rotated bounding boxes to generate general purpose objected proposals. They do a local optimization

over position, size, aspect ratio and rotation from the initial set of sampled windows. In the proposed approach the size and aspect ratio are known and fixed and an exhaustive search over position and rotation is performed. That way the risk of not considering relevant rotations is eliminated. Also, methodology proposed by He et al. [6] is based on hand crafted features assuming that image borders are unlikely to contain objects, and that the object of interest covers a significant part of the interior of the image.

Another approach is to combine segmentation algorithms such as DeepMask [14] and SharpMask [15] with an object detector such as MultiPathNet [23]. This results in object detections augmented with pixel level segmentations. From those segmentations, rotated bounding boxes could be generated or, even better, distance measures could be made using the segmentations directly. However, this approach requires manual pixel level segmentations of training data, while the suggested approach only requires a few manual clicks per training object.

1.2 Experimental setup

Video recordings were made using three Axis M3006-V cameras with a wide field of view, 134 degrees. They were placed in the ceiling at the height of 3.6-meters, pointing straight down to optimise overview over the study area. There was a significant overlap between the camera images to avoid missing events taking place at the border between the cameras. See Figure 1 for some example frames. In total three months of 800×600 video material recorded at 16 Frames Per Second (FPS), was collected.

The cameras were calibrated to compensate for the lens distortion and rectified. Although the physically orientation of the cameras were fixed to make them point fairly straight down, they were still slightly tilted. This tilting was synthetically removed during the rectification. The result of this calibration is video images where the cows have the same size regardless of where in the image they appear. Also, the scan-lines of the three different cameras become aligned, which allows them to be stitched together to form an overview of the entire waiting area c.f Figure 2.

Finally, a Convolutional Neural Network (CNN) was trained to detect the cows in the images, and statistics about how many cows and their distances/relation to each other was extracted. Using that statistical data, scientists working in the field of animal behaviour could form queries to select particular time intervals to watch, such as "show me video clips involving at least two cows with the neck of one cow closer than one meter to the body of the other cow".

2 Camera calibration

The classical pinhole camera model augmented with a lens distortion model was used to model the camera. The camera setup was calibrated by placing markers on the walls and stands in the middle of the waiting area. They were all placed at the same height and thus defined a plane. The mean cow height in the barn was measured, and the plane was placed at their shoulder height. This height was estimated to be 1.49 meters with a standard deviation of 0.05 by measuring twelve random cows in the study area. This is the plane in which all of the landmarks considered below, except for the head, are expected to be found. By projecting detected landmarks back and forth between the camera images and this plane, detections from different cameras can be matched.

In addition to the markers, the focal length and lens distortion parameters were provided by the camera manufacturer. The lens distortion was removed, and a homography was estimated that projected each of the camera images onto the cow shoulder plane. Figure 2 shows a view stitched together from all three images in Figure 1. It shows an overview of the entire waiting area. At the borders between the cameras, the image becomes strange as cows there are viewed from different directions on opposite sides of the border. However, this image is only used for illustration. There is enough overlap between the images to allow them to be processed one by one and then the resulting detections can be combined using this calibration. Figure 3 shows the separate dewarped frames used by the detector. Note how the same cow is almost fully visible in both the left and the middle image.

By using homogeneous coordinates, the pinhole camera model that forms 2D image pixels, $\boldsymbol{x} = (x_1, x_2, 1)$, by projecting world 3D points, $\boldsymbol{X} = (X_1, X_2, X_3, 1)$, using a camera matrix, $\boldsymbol{P}$, can be formed as

$$\lambda \boldsymbol{x} = \boldsymbol{P}\boldsymbol{X} = \boldsymbol{K} \begin{pmatrix} \boldsymbol{R} & \boldsymbol{t} \end{pmatrix} \boldsymbol{X} = \begin{pmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{R} & \boldsymbol{t} \end{pmatrix} \boldsymbol{X}, \tag{1}$$

were f is the focal length of the camera and (p_x, p_y) its principal point while $\boldsymbol{t}$ and $\boldsymbol{R}$ define its 3D position and orientation. This projective image is then distorted using some lens distortion function, $\hat{\boldsymbol{x}} = f_{\text{dist}}\left(\boldsymbol{K}^{-1}\boldsymbol{x}\right)$. Here a radial fish eye lens distortion model is used. By using nominal image coordinates, $\lambda \boldsymbol{x_n} = \boldsymbol{K}^{-1}\boldsymbol{x}$ and $\hat{\boldsymbol{x}}_{\boldsymbol{n}} = (\hat{x}_1 - p_x, \hat{x}_2 - p_y, 1)$ the inverse of this distortion function can be express as

$$|\boldsymbol{x_n}| = \tan\left(\sum_i k_i \left|\hat{\boldsymbol{x}}_{\boldsymbol{n}}\right|^i\right), \tag{2}$$

where k_i is the set of distortion parameters and $|\boldsymbol{x}| = \sqrt{x_1^2 + x_2^2}$. Note that with this distortion model, the focal length becomes part of the distortion and Equation 2 allows the pixels produced by the camera, $\boldsymbol{x}$, to be transformed into nominal projective coordinates, $\boldsymbol{x_n}$, directly. The distortion parameters, k_i, was provided by the manufacturer and the principal point, (p_x, p_y), was assumed to be at the center of the image.

In the camera images, the markers placed in the cow shoulder plane were allocated manually and their distorted coordinates, $\hat{\boldsymbol{x}}$, were registered by clicking on them. Also, real world distances between the marks were estimated using a laser distance meter. From these measurements the world coordinates, $\boldsymbol{x_s}$, were calculated using multidimensional scaling [22]. One homograph, $\boldsymbol{H}$, for each camera was fitted to the point correspondences allowing the pixels

Fig. 1: Example frames from recorded video.

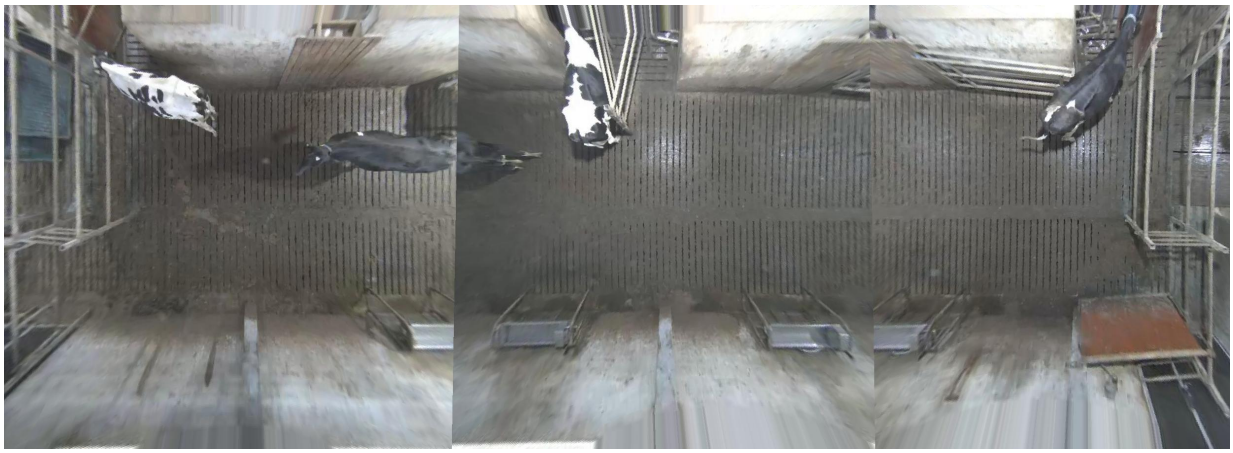

Fig. 2: The frames from Figure 1 projected onto the cow shoulder plane and stitched together

to be projected onto this plane using

$$\lambda \boldsymbol{x_d} = \boldsymbol{H} \boldsymbol{x_n}. \tag{3}$$

This generates a common coordinate system for all of the cameras which allows detections from each of the cameras to be projected into this common frame. That way different cameras could be used in different parts of the waiting area. To minimise the amount of occlusion taking place, each part of the waiting area should use the closest camera to get a view from as straight above as possible. For that, the camera position in the cow shoulder plane needs to be estimated. That can be achieved by RQ factorisation of H [4],

$$\boldsymbol{H} = \boldsymbol{K_d} \boldsymbol{R_d} = \lambda \begin{pmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \boldsymbol{R_d}, \tag{4}$$

were the camera position is given by $\boldsymbol{c} = (c_x, c_y, 1)$. Figure 2 shows an image where each pixel has been chosen from the camera closest to that pixel, i.e. with minimum $|\boldsymbol{x_d} - \boldsymbol{c}|$. The final crops used from each camera consist of the pixels in this stitched view with a border of 75 pixels added on each side to make the overlap 150 pixels, which roughly corresponds to the length of one cow.

3 Data annotation

The complete dataset contains three months of recordings collected from 3 cameras at 16 fps. That makes 400 million frames. Among these, 1722 frames were randomly selected and annotated. The manual annotation was started before the entire video material was recorded. The annotation was performed with a tool that randomly selected a recorded frame and presented it to the human annotator. That means that the 1722 frames are almost uniformly chosen over time, but not exactly, as the frames in the beginning of the sequence have a slightly higher probability to be included. The effect is however minor. To verify that, the mean distance between two adjacent annotated frames was measured to be 30 min. So the annotated frames are believed to be fairly uncorrelated and should give a good representation of the distribution of frames that could be expected to be seen from these cameras.

The annotated subset contained in total 6399 cows in the images. Each cow was annotated with seven landmark points: head, left and right shoulder, front middle, left and right hip and back middle. In addition to that one additional landmark "cow centre" was defined as the mean of front middle and back middle. This data was then used to train a CNN detector.

4 CNN cow detector

The detector was split into two steps. The first step (the landmark CNN) is a fully convolutional CNN that detects the landmarks in the image. The second step (the cow CNN) is another CNN that uses the probability map produced by the first CNN as input and detects the cows and their orientation.

4.1 Landmark CNN

Only four of the landmarks were used: head, front middle, cow centre and back middle. The discarded landmarks left and right shoulder are close to front middle while the discarded landmarks left and right hip are close to back middle. This was done due to the fact that these landmarks do not provide much extra information about the cow's position or orientation and including them would slow down the experiments. Also, due to the max-pooling used in the CNN, the output feature map has too low resolution to separate such close landmarks well, which means that they might compete for the same pixels in the output probability map. That might actually harm performance.

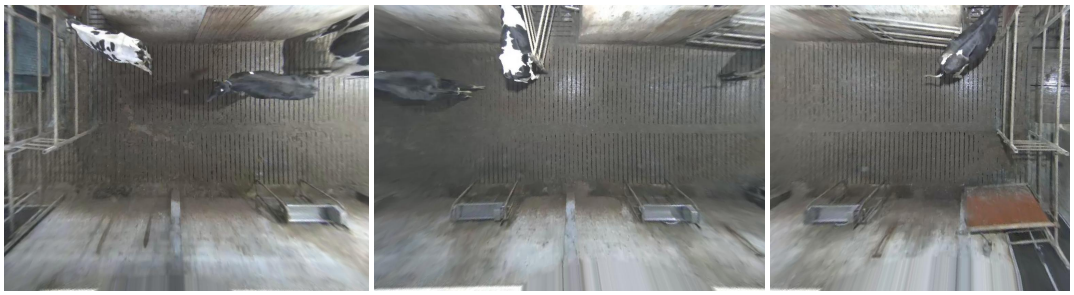

Fig. 3: Dewarped frames from each of the cameras with overlaps to allow the detector to process them one by one.

Layer type	Size	Channels
Conv + BNorm + Relu	3x3	32
MaxPool(stride=2)	2x2	
Conv + BNorm + Relu	3x3	64
MaxPool(stride=2)	2x2	
Conv + BNorm + Relu	3x3	128
Conv + BNorm + Relu	3x3	128
MaxPool(stride=2)	2x2	
Conv + BNorm + Relu	3x3	256
Conv + BNorm + Relu	3x3	256
MaxPool(stride=2)	2x2	
Conv + BNorm + Relu	3x3	512
Conv + BNorm + Relu	3x3	512
MaxPool(stride=2)	2x2	
Conv + BNorm + Relu	1x1	1024
Conv + BNorm + Relu	1x1	1024
Conv + Softmax	1x1	5

Table 1 CNN architecture for landmark detection. The input is an image of any size with 3 rgb channels scaled to the range $[0,\ 1]$. The output is probability map segmenting the entire image into 5 classes: Ground, Cow front middle, Cow center, Cow back middle and Cow head.

However, extending to use more landmarks is straightforward. The architecture of this network is a fully convolutional CNN similar to VGG [20] with batch normalisation [8] after each convolution step. That means that instead of fully connected layers at the end, 1×1 convolutions are used. Details are shown in Table 1. Only valid outputs from the convolutional and maxpool layers are kept.

Applying the network to a high resolution image will apply the landmark detector to every position in the image, resulting in an effective implementation of a sliding window detector. It can be applied to images of resolution $118 + 32n_w$ times $118 + 32n_h$ and will produce an output image of resolution $n_w \times n_h$ with five channels. Each of the channels contains the detection probability of each of the five classes (four landmarks and one background class). A pixel at (x, y) in the output probability map corresponds to a landmark detection at position $(32x + 75, 32y + 75)$ in the input image.

During training, $n_w = n_h = 1$ was used and the net was trained on patches of 150×150 pixels extracted from the input images. The positive examples were centred on the landmarks and randomly jittered ± 16 pixels (as the distance between output pixels is 32 input pixels). Negative patches were selected at centres more than 32 pixels from any landmark. In addition to the positive and negative patches a set of do not care patches was selected at random centres at distances between 16 and 32 pixels from landmarks. The ground truth probability of these patches belong to the class of the landmark was set to 0.5, and the probability that they are ground was set to 0.5. In some cases, several landmarks appear within 32 pixels of the patch centre. In that case, the probability mass was distributed uniformly among all involved classes. Also, all pacthes were randomly rotated ± 180 degrees.

The weights of the convolutions were initiated using random samples drawn from a Gaussian distribution truncated at 2σ, with standard deviation $\sigma = \sqrt{\frac{2}{n}}$, where n is the number of inputs[5]. The networks were regularised with weight decay of 0.0001 and optimised using stochastic gradient descent with 0.9 momentum. The learning rate was initiated to 1.0 and reduced by a factor 10 each time the validation error flattens. This was achieved by periodically, during the training, evaluating the CNN on the validation data (10% of the annotated data that was not used for training) and when the validation error rate stopped decreasing, the learning rate was reduced.

Once the net was trained, the step length of the last maxpool layer was reduced from two to one to increase the output resolution. After that pixels at (x, y) in the probability maps correspond to detections at $(16x + 75, 16y + 75)$ in the input image. During testing n_w and n_h depend on the size of the input frame which varies from camera to camera. Figure 3 show three different input examples, one from each camera.

Layer type	Size	Channels
MaxPool(stride=1)	3x3	
Log		
Conv + BNorm + Relu	13x13	33
Softmax		

Table 2 CNN architecture to detect cows and their orientation. The input is the 5 channel probability map from the landmark detector with the last MaxPool removed to increase resolution. The output is a probability map that segments the image into either background or cow in one of 32 different orientations.

4.2 Cow CNN

The second step is another fully convolutional CNN that works with the probability map produced by the first CNN as input and tries to detect the cows and their orientations. The full circle is divided into 32 equally spaced orientations which generate 32 different oriented cow classes. In addition to that, there is the "no cow" class, which makes the total number of classes of this CNN 33. The input probabilities were turned into log likelihoods as it makes more sense when summing them together. Then the network consists of a single 13×13 convolutional layer. Details are shown in Table 2. The same resolution was kept after each layer which means the relationship between pixels coordinates in the output map and the original input frame is the same as for the landmark CNN.

The net was then applied to the fully rectified training images producing probability maps of $44 \times 46 \times 5$ pixels. These were used as training examples for the cow detection net (without splitting them into patches). Random rotations ± 180 degrees and translations ± 16 pixels were applied to both input images and their annotations. This means that all the 6399 annotated cows will eventually be used as a positive example to each of the 32 orientation classes. Output ground truth probability maps of $44 \times 46 \times 33$ pixels were constructed from the annotations by projecting each cow, i, center point into the probability map as (x_i, y_i) and calculate its angle a_i as the angle of the line between front middle and back middle landmarks. Then a binary $44 \times 46 \times 33$ mask $B\left(x, y, c\right)$ is formed, containing a background mask

$$B\left(x, y, 32\right) = \begin{cases} 0 & \text{if} \quad \begin{array}{r} \lfloor x_i \rfloor \leq x \leq \lceil x_i \rceil \\ \lfloor y_i \rfloor \leq y \leq \lceil y_i \rceil \end{array} \\ 1 & \text{otherwise} \end{cases} \tag{5}$$

and 32 orientation masks

$$B\left(x, y, c\right) = \begin{cases} 1 & \text{if} \quad \begin{array}{r} \lfloor x_i \rfloor - 1 \leq x \leq \lceil x_i \rceil + 1 \\ \lfloor y_i \rfloor - 1 \leq y \leq \lceil y_i \rceil + 1 \\ \text{adist}\left(\frac{2c\pi}{32}, c_i\right) < \frac{2\pi}{32} \end{array} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

for $0 \leq c \leq 31$ and all i. The adist function calculates the absolute angular distance between two angles. The ground truth probability masks are then produced by normalising B to sum to 1 for each pixel. Finally, the network is trained using the same hyper parameters as for the landmark CNN.

5 Watchdog evaluation

5.1 Cow detection

To evaluate the system, 6400 frames spread over the entire recording were processed by the CNN. It was not ensured that none of the training frames were chosen here, but since they both were choosen randomly from a set of 400 millon frames, the chance that they are mutually exclusive is 97.2%. A simple watchdog extracting frames

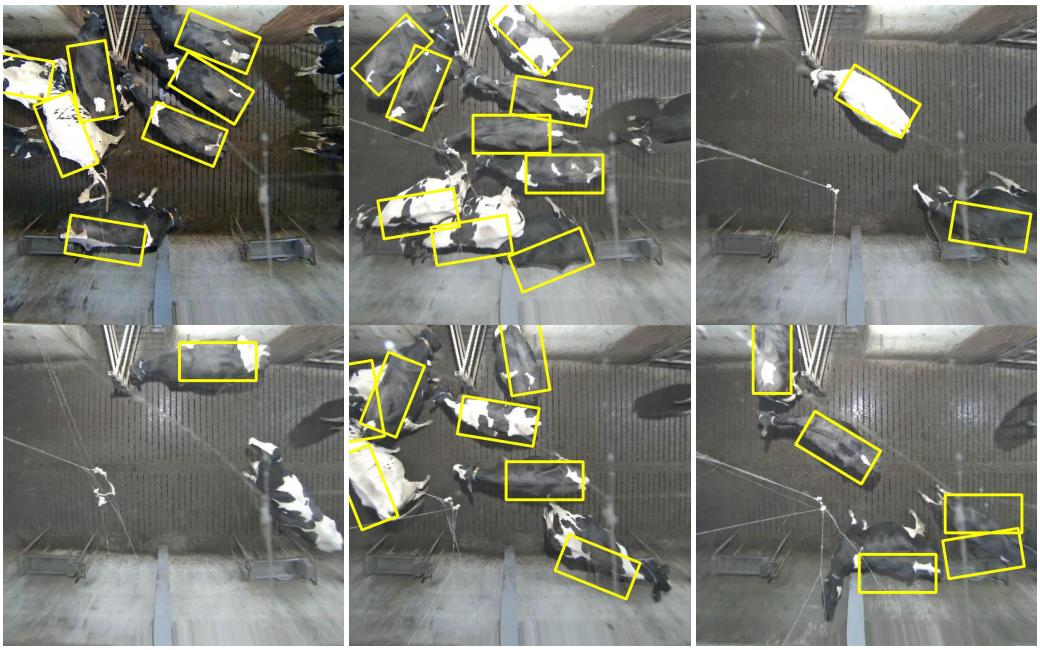

Fig. 4: Top row: Three images correctly interpreted (all cows detected and no extra detections). Bottom row: The three images were the errors were made (1 missed cow and two extra detections).

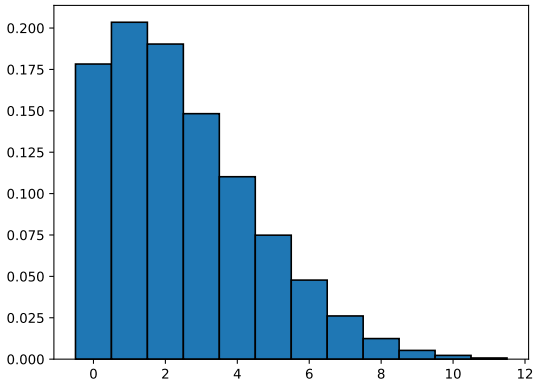

Fig. 5: A histogram of the number of cows detected per frame. Frames with 0 or 1 cows are uninteresting, which according to this detector is 38% of the frames.

containing two or more cows was implemented. That would be the most basic requirement for interaction, and already this simple criteria, discarded 38% of the recordings. To verify that the discarded video was uninteresting, 500 random frames selected by the watchdog and 500 random frames discarded by the watchdog were automatically annotated using the CNN results and studied manually. Cows intersecting the borders were ignored in the sense that the images were considered correct regardless of whether such border cases was detected or not. A detection was considered correct if its rotated bounding box overlapped more than 50% of the cow back.

Most, 94.5%, of the images were perfectly interpreted, i.e. all cows present were detected and no extra detections. Most of the errors were made in crowded situations where more than two cows were present and detected and thus classified correctly according to the watchdog criteria. Only a single case was found were the watchdog erroneously discarded a frame, resulting in a watchdog accuracy of 99.9%. Note that this is a frame accuracy and an interaction consists of multiple frames, so missing an entire interaction would be extremely unlikely even though the frames within that interaction would be somewhat correlated. In total, those 1000 images contain 2041 cows. 16 of those were not detected and 48 extra detections were made yielding a cow hit rate of 99.22% with a false alarm rate of 2.35%. Some example detections are shown in Figure 4. Two of the reasons for mistakes are inter-cow occlusion and the combination of landmarks from different individuals. A histogram of the number of cows detected per frame is presented in Figure 5.

The evaluation runs at 6.55 fps on a single Tesla K20m GPU, using single precision floats.

5.2 Long term

Half a year after the original recordings were made, another month of video material was collected, c.f Figure 6, and the original detector was tested on this data. It did however not perform very well. A set of 931 frames were sampled from the new recording and passed to the detector. It produced annotated images that were then inspected manually. Among these, 510 or 55% of the images were perfectly interpreted in the same sense as above, while 421 or 45% contained some sort of mistake. A lot of false detections were made. Especially in over exposed areas and there were also significantly more misses. In a random sample consisting of 50 of the 421 erroneous frames there was in total 128 cows correctly detected, 28 extra detections made and 45 cows missed.

Two main differences between this new data and the old one have been identified. First, the cameras have become quite dirty. They were cleaned a few times during the experiments, but they became dirty quite fast, so for a system like this to become useful it will have to be able to handle somewhat dirty cameras. Second, the sun was low in the sky during the time of year of the new recording. That results in a different lighting situation. It is an indoor scene, but there are windows letting in the sun light.

From this new data, the 421 frames where the old detector made errors as well as 3 frames that contained no errors were manually annotated in the same way as before. They contained in total 2880 cows. The detector was retrained using both the new and the old data, which resulted in a detector with more even performance across the varying seasons. A validation set consisting of 10% of the annotated data (196 frames) was separated out and not used during the training. This set was used to evaluate the new detector. In total it

Fig. 6: Example frames from a second recording made half a year after the first one.

contained 408 cows, and 396 of these were detected with only 6 extra detections. That gives a cow hit rate of 97% with a false alarm rate of 2.9%. In terms of frames 92.8% were perfectly interpreted while there were mistakes made in 14 of the frames. Among those 14, 13 contained more than one cow and thus the hit rate of the watchdog finding frames with two or more cows were 99.5%.

5.3 Interaction detection

To remove even more of the uninteresting video, an additional feature, the minimum distance between the cows in the scene, was also extracted. Then a short sequence containing a lot of interesting interactions consisting of 187 frames uniformly sampled over 10 minutes was extracted and manually annotated by an expert. Five different interactions were manually identified: body pushing, butting, head butting, head pressing and body sniffing. Frames where any of these interactions were present, were considered interesting and all other frames uninteresting. The minimum cow distance, d_f, for each frame f, was extracted. It is believed to be a good feature as cows need to be close to interact. Also, the cows needs to be close for some period of time, so we take the maximum over 9 consecutive frames and use as feature, $x_f = \max_{f-4 \leq i < f+4} d_i$ for frame f.

By thresholding x_f a simple detector is formed. By choosing different thresholds, the results could be varied between not detecting any uninteresting frames and not missing any interesting frames. The per frame ROC curv in Figure 7 shows the amount of the interesting frames detected (true positives) as a function of the amount of the uninteresting frames remaining (false positives) for different thresholds. It is, for example, possible to discard 20% of the uninteresting frames while only losing 3% of the interesting once. If it is acceptable to lose more of the interesting frames, even more of uninteresting frames could be discarded as detailed by the curve.

An interaction will consist of multiple frames, and in many situations it is enough to detect a single one of those frames as the user then could be allowed to also look at adjacent frames. Figure 7 also contains a per interaction roc curve that shows the amount of the interactions for which at least one frame was detected (true positives) as a function of the amount of the uninteresting frames remaining (false positives). It shows that it is for example possible to discard 35% of the uninteresting frames without losing any of the interactions.

Note that this is in addition to the filtering by the number of cows present, performed in Section 5.1. By combining the two filters, 50% of the uninteresting frames could be discarded while only loosing 4% of the interesting frames. If one frame per interactions is enough it would be possible to discard 60% of the video without losing any interactions.

5.4 Evaluation summary

The system was evaluated on several levels: cows, frames and interactions. Single cows were detected with a hit rate of 97% and a false alarm rate of 2.9%. Single frames were perfectly interpreted 92.8% of the time. Frames with only zero or one cow present were correctly identified 99.5% of the time. All interactions were detected while discarding 60% of the video as uninteresting.

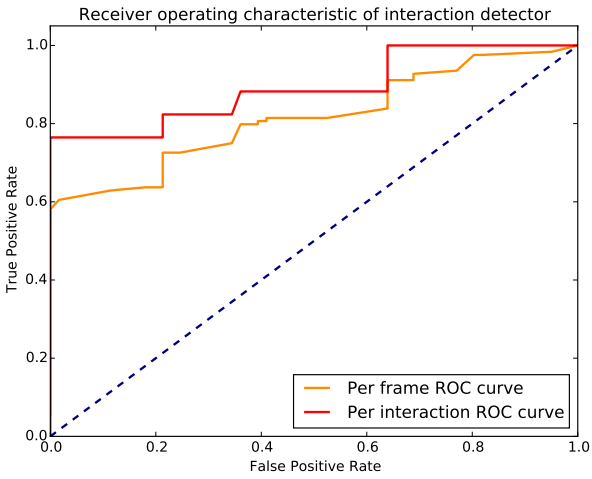

Fig. 7: Results for different thresholds, with the amount of interesting frames (orange) or interactions (red) kept (true positive rate) plotted as a function of the amount of uninteresting frames kept (false positive rate).

6 Conclusions

A CNN cow detection system has been developed. It can detect and count the cows present in the image with high precision. 92.8% of the test images were perfectly interpreted in the sense that the system was able to place a rotated rectangle on each cow and nowhere else. This detector was used to discard 50% of the recorded video as uninteresting while only losing 4% of the interesting video. If detecting a single frame per interaction is enough, 60% of the test dataset could be automatically discarded without losing any of the interactions. Regarding detection of single cows, the hit rate was 97% with a false alarm rate of 2.9%. Note that these numbers depend on how large portion of the recorded video is actually interesting and how often the scene is crowded (which is when most mistakes are made). This will vary between different farms and studies.

Another important conclusion was that the even though it is an indoor scene, the lighting variations caused by the varying seasons are significant. It was not enough to collect data from three months. Instead both training data and evaluation data from different seasons are needed to build a system that can operate all year around. This seems to concur with the findings of Porto et al. [17] who write that their detector, CVBS, “were highly affected by the shadows” and that “worst operating conditions for the CVBS occurred in the summer”.

7 Acknowledgement

The computations were performed on resources provided by the Swedish National Infrastructure for Computing (SNIC) at Lunarc. The Swedish Research Council for Environment, Agricultural Sciences and Spatial Planning (FORMAS) is acknowledged for the funding of the project. The farmer, Mikael Palm, is acknowledged for the use of his diary barn and cows during the pilot study.

8 References

1 Khalid Abdul Jabbar, Mark F Hansen, Melvyn Smith, and Lyndon Smith. Locomotion traits of dairy cows from overhead three-dimensional video. *Visual observation and analysis of Vertebrate And Insect Behavior*, 2016.

2 T. J. DeVries, M. A. G. von Keyserlingk, and D. M. Weary. Effect of feeding space on the inter-cow distance. *Aggression, and Feeding Behavior of Free-Stall Housed Lactating Dairy Cows. J. Dairy Sci*, 87, 2004.

3 O Guzhva, H Ardö, A Herlin, M Nilsson, K Åström, and C Bergsten. Feasibility study for the implementation of an automatic system for the detection of social interactions in the waiting area of automatic milking stations by using a video surveillance system. *Computers and Electronics in Agriculture*, 127:506–509, 2016.

4 R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

5 Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *CoRR*, abs/1502.01852, 2015.

6 Shengfeng He and Rynson W.H. Lau. Oriented object proposals. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

7 P.. Hemsworth. Human-animal interactions in livestock production. *Appl. Anim. Behav. Sci*, 81, 2003.

8 Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

9 R. J. Kilgour. In pursuit of 'normal': A review of the behaviour of cattle at pasture. *Appl. Anim. Behav. Sci*, 138, 2012.

10 Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*, pages 21–37. Springer, 2016.

11 P. Martin and P. Bateson. *Measuring Behaviour An Introductory Guide*. Cambridge University Press, Cambridge, 2007.

12 Carlos A Martinez-Ortiz, Richard M Everson, and Toby Mottram. Video tracking of dairy cows for assessing mobility scores. 2013.

13 E. S. Nadimi, R. N. Jörgensen, V. Blanes-Vidal, and S. Christensen. Monitoring and classifying animal behavior using ZigBee-based mobile ad hoc wireless sensor networks and artificial neural networks. *Comput. Electron. Agric*, 82, 2012.

14 Pedro H. O. Pinheiro, Ronan Collobert, and Piotr Dollár. Learning to segment object candidates. *CoRR*, abs/1506.06204, 2015.

15 Pedro H. O. Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. *CoRR*, abs/1603.08695, 2016.

16 A. Polikarpus, T. Kaart, H. Mootse, De Rosa, Arney G., and D. Influences of various factors on cows' entrance order into the milking parlour. *Appl. Anim. Behav. Sci*, 166, 2015.

17 Simona MC Porto, Claudia Arcidiacono, Umberto Anguzza, and Giovanni Cascone. The automatic detection of dairy cow feeding and standing behaviours in free-stall barns by a computer vision-based system. *Biosystems Engineering*, 133:46–55, 2015.

18 Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

19 Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

20 K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

21 Porto S.M.C., Arcidiacono C., Anguzza U., and Cascone G. Detecting cows at the feed barrier by means of an image analysis algorithm. *International Conference RAGUSA SHWA 2012, September 3-6, 2012, Ragusa - Italy*.

22 Gale Young and A. S. Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3(1):19–22, 1938.

23 Sergey Zagoruyko, Adam Lerer, Tsung-Yi Lin, Pedro H. O. Pinheiro, Sam Gross, Soumith Chintala, and Piotr Dollár. A multipath network for object detection. *CoRR*, abs/1604.02135, 2016.