Computers and Electronics in Agriculture 178 (2020) 105761

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Video analytic system for detecting cow structure

He Liu, Amy R. Reibman, Jacquelyn P. Boerman

Purdue University, 501 Northwestern Ave, West Lafayette, IN 47907, USA

ARTICLE INFO

Keywords:
Video analytics
Pose estimation
Cows
Video processing

ABSTRACT

In animal agriculture, animal health directly influences productivity. For dairy cows, many health conditions can be evaluated by trained observers based on visual appearance and movement. However, to manually evaluate every cow in a commercial farm is expensive and impractical. This study introduces a video-analytic system which automatically detects the cow structure from captured video sequences. A side-view cow structural model is designed to describe the spatial positions of the joints (keypoints) of the cow, and we develop a system using deep learning to automatically extract the structural model from videos. The proposed system can detect multiple cows in the same frame and provides robust performance for the body region under practical challenges like obstacles (fences) and poor illumination. Compared to other object detection methods, this system provides better detection results and successfully isolates the body keypoints of each cow even when the cows are close to each other.

1. Introduction

Monitoring health is a critical component of animal agriculture, because healthy animals are more productive. Such monitoring is often performed visually, because animal appearance and behavior are key indicators of health changes. For example, trained farm personnel can analyze a dairy cow's health condition based on visual appearance (Fleishman and Endler, 2000), and can detect potential illnesses such as lameness (Cook, 2020). However, time and labor limitations preclude a human routinely watching for these changes, especially in commercial farms which house a large number of cows.

Thus there is increasing interest in supplementing human observations with automated video analytics for animal agriculture (Pluk et al., 2010; Zhao et al., 2018; Poursaberi et al., 2010; Condotta et al., 2018; Leonard et al., 2019). The primary focus for cows has been to detect lameness (Cook and Nordlund, 2009). The methods typically focus on a specific body region instead of the entire cow body and its structure. For example, back curvature is detected in Poursaberi et al. (2010) and Viazzi et al. (2013), while trajectories of the legs and hooves are tracked in Song et al. (2008) and Zhao et al. (2018). Other applications for cows include tracking (Ter-Sarkisov et al., 2017), behavior analysis (Guzhva et al., 2016), cow identification (Andrew et al., 2017; Zhao et al., 2019; Shao et al., 2019) and body score estimation (e.g. Spoliansky et al., 2016).

One drawback of these previous methods is that their processing techniques are developed for a specially-designed environment where the captured images are clear enough to process. For example, often each cow must stand or walk individually on a well-lit pathway with a clear background and no obstructions. To be applicable in the less constrained environments typical of those in commercial dairies, these methods require an additional fundamental step to detect and locate the cows within the images or videos.

Automated methods to localize an object (e.g., a cow) in an image or video are termed object detection(Tsai et al., 2016), object segmentation (Lee et al., 2011; Redmon and Farhadi, 2018), or semantic segmentation (Wang et al., 2015; He et al., 2017; Chen et al., 2017). The goal of these analytic methods is to generate a binary mask indicating the location of the identified objects and their labels. Methods can be applied to either individual images (He et al., 2017; Chen et al., 2017; Redmon and Farhadi, 2018; Maninis et al., 2018) or to a video (Wang et al., 2015; Tsai et al., 2016; Lee et al., 2011; Caelles et al., 2017; Tokmakov et al., 2017; Cheng et al., 2017; Voigtlaender and Leibe, 2017). While these methods have been designed to localize a wide range of different types of objects, with appropriate training they can also be tailored to a specific task like detecting cows (see for example the experiments below).

The above methods all generate bounding boxes or pixel-level masks to represent detected objects, without identifying any structural information of the object. However, to assess characteristics of an animal (i.e. body size or gait), simply having a binary mask that labels the cow's location is inadequate. Further information of the cow's structure is required, such as the locations of all body parts or joints. In video analytics terminology, these body locations and joints are called keypoints. Methods to locate them in isolation are call keypoint detection

E-mail addresses: liu1433@purdue.edu (H. Liu), reibman@purdue.edu (A.R. Reibman), jboerma@purdue.edu (J.P. Boerman).

https://doi.org/10.1016/j.compag.2020.105761
Received 14 February 2020; Received in revised form 25 July 2020; Accepted 30 August 2020

algorithms, and pose estimation detects keypoints as well as connecting structural information. Significant progress has been made in human pose detection (Toshev et al., 2014; Newell et al., 2016; Newell et al., 2016; Cao et al., 2017; Cao et al., 2017, 2018), by leveraging several public human pose datasets (Andriluka et al., 2014; Lin et al., 2014).

Recently, keypoint detection and pose estimation has also been applied to animals (Mathis et al., 2018; Pereira et al., 2019; Günel et al., 2019). These methods provide a means for users to define body parts; this allows the algorithm to adapt to different animal structures. The DeepLabCut toolbox (Mathis et al., 2018) also provides simple access to fine-tune the underlying convolutional networks. In addition, with a small amount of training data, it achieves promising results when applied to video sequences that have been captured under laboratory conditions, that contain a single animal per image in front of a clean background with good illumination. However, we will show below that it performs less well when applied to videos of cows walking in a cow barn that have been recorded in a less controlled setting.

In this paper, we present a video analytics system to detect the keypoints of a cow and the associated connecting structure. Combining deep learning with domain knowledge about cows, this system is designed to address several challenges that exist when processing videos captured on a practical dairy farm: more than one animal, poor lighting conditions, and additional objects like occluding fences. In particular, our system estimates the number of cow objects in a frame and detects the body parts of every individual cow. For each cow object, the detected keypoints are composed into a structural model of the side-view of a cow, which describes the spatial location of the cow, the body contour, the positions of major joints, and the trajectories of their movement. This detailed information provides interpretable knowledge for further health analysis; for example, weight can be estimated using distances between joints on the cow's body (Song et al., 2018). More detail on the structural model is discussed in Section 2.1 below.

One advantage of our system is that it is designed to operate on videos captured on a commercial farm without interrupting the daily operation of the farm. In such a scenario, the environment in which video is captured cannot be fully controlled. Camera positions and angles are limited, lighting and obstacles like fences are governed by the needs of the farm, and the forward movement of cows are unconstrained. Given these limitations, our system uses only a surveillance camera and no specialized hardware. Cows are observed during the daily farm activities, and the system accommodates obstructions like fences and image degradations like poor contrast from weak illumination (Kawakatsu et al., 2017).

The main contributions in this work are highlighted as follows. First, we design a cow model with keypoints that presents structural information about a cow that would be necessary for subsequent cow health analysis. Second, a system is developed to extract the cow structural models from videos that are captured from practical dairy farms, with multiple cows and occluding fences. Third, for the cow structural model, we also develop additional evaluation metrics that operate with limited ground-truth labels. Fourth, we demonstrate experimentally that this system outperforms other popular object detection algorithms when presented with practical challenges. Finally, in later experiments, we use multiple video datasets to demonstrate the robustness of the proposed detection system to different cameras.

This paper is organized as follows. Section 2 introduces the proposed cow structural model including the keypoints and their spatial constraints. Section 3 presents the cow structure detection system, followed by detail explanations of the detection module and the post-processing module. The related cow structure evaluation metrics are described in Section 4. Next, Section 5 describes the on-farm video data collection and preparation, as well as three experiments of our detection system. The first experiment evaluates each individual component of our system; the second explores the robustness of our system using three sets of video data; and the third experiment demonstrates the advantages of our proposed system relative to other popular object segmentation methods. Finally, Section 6 summarizes this work.

2. Structural cow model

In this section, we first introduce the keypoints in our cow structural model in detail, and then describe the spatial constraints between the keypoints. These constraints are further used in the detection system for separating multiple cows and detecting missing parts.

2.1. Cow body keypoints

This proposed structural model is designed to represent a detected cow object in the frame more effectively than using a binary cow mask. It is designed to provide both the spatial location and cow structural details, such as the body shape and positions of the body parts. For consecutive video frames, this model should also provide information so that we can track the movement or motions of these body parts. Inspired by recent approaches to model the human skeleton (Toshev et al., 2014), we combine some anatomical cow joints with other spatial keypoints to represent the cow pose, and the cow structural model is built by connecting the keypoints.

Fig. 1 shows our proposed side-view cow structural model. There are 17 points in total to describe the important locations of a cow object from this angle. The upper body region has 9 points, including the head region (blue) and the main body region (red). Connecting these points forms the contour of the upper body region (green lines). Another 8 points are in leg-hoof regions which represent the four limbs, and each limb has a pair of leg and hoof joints. Comparing to the anatomical 51-point cow skeleton model (Aujay et al., 2007), we only select visually-observable joints. Some joints such as the elbow and stifle joints are neglected because their positions are not readily visible and thus difficult to isolate visually. In addition, we also add some points such as the two bottom corners H and I show in Fig. 1. Even though they are not physical joints, connecting them with other joints forms a closed contour which spatially locates the body region. The point E on the spine is also an added point, because connecting three spine points provides information about the back curvature which is useful for lameness detection.

There are two general observations about the keypoints in this cow structural model. First, the points in the main body region (red in Fig. 1) are always visible from the side-view, and their relative spatial locations do not change dramatically when the cows are walking. Second, the leg and hoof points are much more difficult to detect compared to

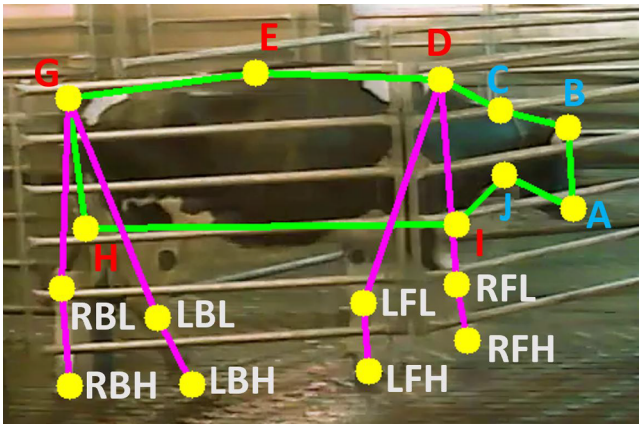


Fig. 1. The proposed cow structural model. 4 blue head region points: A:nose, B:head, C:top of neck, J:bottom of neck. 5 red body region points, D:shoulder, E:spine, G:tailhead, H:mid-thigh, I:bottom of shoulder. 8 white leg and hoof points, with name format: Right/Left + Front/Back + Leg/Hoof. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the upper body region points because of the practical issues such as bad illumination, shadows, and fast leg movement. Distinguishing between the points from the left or the right leg is also difficult when there are obstacles in front, for example the horizontal fences shown in Fig. 1.

2.2. Keypoint constraints

Practical constraints limit the potential relationships among the keypoints in both space and time. The camera, located at a fixed position on the side wall, always captures a side-view of the cow as it walks through the fences. These fences allow no more than one cow to walk through at a time. In this case, each cow shown in the video is facing the same direction, and the keypoints of each upper body region are always located at relatively fixed spatial positions. For example, a cow's head always appears on the right side of its body, and the body does not change size. As a result, we can compute general relationships that constrain the keypoints in the cow structural model.

To model the constraints, we first define the center of the cow's body. This center point is computed as the spatial center of all the keypoints from the cow's upper body region. Note that the points in the leg-hoof region are not used to compute the center point because their positions are not relatively fixed when the cow moves. Then we can estimate the relative spatial relationship between the center and all the keypoints.

Fig. 2 visualizes the keypoint constraints. The middle X shows the cow center c, and the relative spatial locations of the upper body parts appear surrounding the center. Each body part mapping function F_j is based on two Gaussian probability distribution in both the horizontal and vertical directions; the distribution parameters are estimated for each direction individually using manually-labelled keypoint positions. Given a fixed center point, the probability of the mapped keypoints are shown as an ellipse in Fig. 2.

Formally, for a fixed cow center point $c^* = (x, y)$, we define a set of mapping functions $F_j(\cdot)$ that describe the relative spatial locations of every upper body-part point p_j^* to the center,

$$p_j^* = F_j(c^*) \tag{1}$$

where j is the index of the body part. Each mapping function F_j is characterized by a 2D Gaussian model, and the parameters are trained using all ground-truth labels. During the training process, the approximate cow center c is computed first as the geometric center of all labelled body parts. This center point has no physical meaning, but provides a estimated position for the cow in the video frame. Next the parameters in each F_j are estimated individually based on their relative spatial locations to c. Each keypoint has a different mapping function, and keypoints that are closer to the center can be more accurately estimated.

In the next section, we show how these constraints can be used to separate cows which are spatially close together in the frame. They also provide a reference when assigning body-part candidates to each individual cow object in the post-processing module.

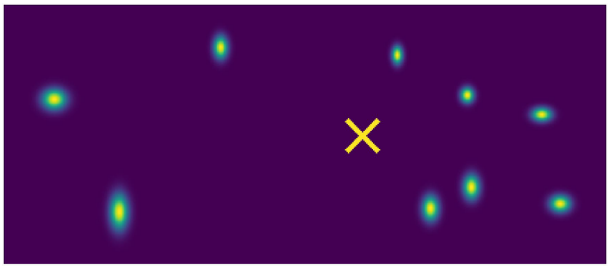


Fig. 2. The constraints between the upper body keypoints of the cow structural model. The yellow X is the cow center, and the surrounding points shows the relative positions of the keypoints in the upper body region. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3. Skeleton detection system

This section introduces our proposed system to detect the structure of cows. We first review one popular work for keypoint extraction and then describe the components of our proposed system. Then we explicitly introduce two main processing components: the body part extraction module and the post-processing module.

3.1. The DeepLabCut toolbox

The DeepLabCut toolbox (Mathis et al., 2018) is a recent popular method to extract keypoints from video sequences. The inputs are color images from videos, and it applies a CNN to generate confidence maps that represent the potential keypoint locations. One advantage of the DeepLabCut toolbox is that it provides simple access for users to manually define the output body parts, and the toolbox automatically alters the last layer of the CNN based on the number of body parts. For example, there are 17 confidence maps generated in our case because we have 17 keypoints in our cow structural model. In our system, we apply the network created by the toolbox to extract the keypoints of our cow structural model. The detailed extraction process is explained latter.

However, other modules from the toolbox are less suitable for our application because of two major limitations. First, this platform is designed and evaluated with videos captured from a laboratory environment with clear objects and background. But our cow videos, generated from a commercial farm, have low video quality and the view of the cows are often blocked by obstructions. Later experiments show that the original DeepLabCut does not provide robust detection results on our videos. Second, this method assumes there is only one object in a frame, so it only chooses one body part from each confidence map. If there are multiple body-part candidates detected, only the position with the highest confidence score will be selected. But in videos generated from commercial farms, there could be multiple cows and obstructions like fences that easily cause false detection. We address these two limitations and build a general keypoints detection system which extracts robust keypoints on our cow videos.

3.2. Proposed system

This detection system is targeted to extract the structural model for every cow object from video sequences. Fig. 3 presents the overall system; its primary components are two CNNs for the extraction and a post-processing module. The body part extraction module uses trained networks to convert each single image into a group of confidence maps. Each map shows the potential locations of a particular body part, and the values of the map represent their detection confidence. The post-processing module generates the final structural model based on two groups of confidence maps and the trained keypoint constraints. Both modules are discussed in detail in the next two sections.

In this figure, both the training process and the testing process are labelled using colored arrows. During the training process (indicated by the green arrow in Fig. 3), the ground-truth labels are used to fine-tune both CNNs and the keypoint constraints. During operation (indicated by the yellow arrow), the system takes the input of both the color image and the frame difference image on the left, and generates the cow structural model for a single image. The frame differences are the absolute pixel value difference between adjacent frames, which highlights the moving cows and other temporal information in the video. After all the frames from a video sequence are processed, the post-processing module refines all the detected cow structures based on temporal information.

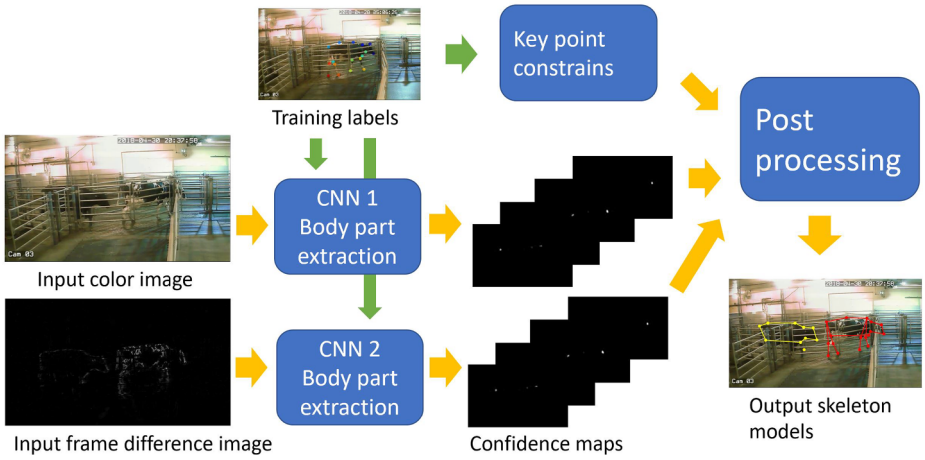


Fig. 3. A diagram of the proposed system. The green arrows show the training process and yellow arrows present the process during operation. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.3. The body part detection module

The goal of this module is to find the spatial locations of all potential keypoints from raw images. In our system, we apply the original DeepLabCut network (Mathis et al., 2018), labeled CNN1, to extract keypoints from color images. This network structure follows DeeperCut (Insafutdinov et al., 2016), and is implemented using ResNet (He et al., 2016) for the convolution stages, followed by one de-convolution layer before the output layer to recover the target spatial locations of the keypoints. The last two convolution layers apply atrous convolution, which increases effective fields-of-view of the applied convolution and preserves spatial resolution (Chen et al., 2017). By default, the DeepLabCut network is pre-trained on ImageNet (Krizhevsky et al., 2012) for image classification tasks, and we use our own cow labels to fine-tune the last de-convolution layer for keypoint detection. During the fine-tuning process, the intermediate layers are fixed, and they extract the spatial features from the input. For the last upsampling layer, we first adjust it to produce the specific number of cow keypoints, and we train this layer with our labelled cow data. In the experiment, the training and testing frames are randomly chosen using two guidelines. First, adjacent frames are not selected because they are too similar spatially. Second, we try to include frames where the cows appear in different spatial locations. This provides a varied training dataset which leads to robust system performances.

However, as mentioned above, low video quality and heavy obstacles influence the performance. To overcome this issue, we add an extra network, CNN2, into the system. The architecture of this network is same as the first, but it processes frame difference images. There are three major advantages of using frame difference images for our cow videos. First, because we have fixed cameras, the frame difference image better captures the moving objects and eliminates the stationary obstacles such as fences. Second, many of our target keypoints are on the contour of the cow body, and the frame difference highlights these edges of a walking cow.

Third, frame difference also reduces the influence of color variation. This is useful, because the color responses of different cameras are not the same especially under poor illumination. In addition, the majority of the cows have color variations introduced by the patterns on the cows, but some cows only have a single coloring, such as pure white, black or brown. If these patterns are not included in the training frames, then the color-based CNN methods would likely fail to detect cows with unseen colors. As a result, using frame differences provides robustness to these factors.

However, using the frame difference images alone is not enough because they eliminate too much spatial information, especially for legs and hooves. This is because most of the legs are stationary even when the cows are moving. As a result, our system merges both networks together to improve the body part detection accuracy.

3.4. The post-processing module

The post-processing module collects and merges the confidence maps from the two CNNs, and assigns the cow body-part candidates to each cow object instead of just to one cow per frame. This step is critical to our system, because incorporates domain knowledge about cows into the estimation of keypoint locations by incorporating constraints about how a cow may move. These constraints help improve the accuracy of estimated keypoint locations for both the body region and the leg and hoof region, but its performance on the body points is better than on the leg and hooves because there are more constraints on the body than the legs. In addition, this step enables the system to detect multiple cows in the same image frame and track their temporal movements. There are three major steps in this post-processing module: body part extraction, spatial clustering, and temporal filtering.

3.4.1. Body part extraction

This step extracts the spatial locations of all body-part candidates from the confidence maps generated by the CNNs. At this stage, the number of cow objects in the image is unknown and we want to extract all possible candidates. For each body part, we use non-maximum suppression (Neubeck and Van Gool, 2006) to select only those pixel positions whose confidence scores are higher than their neighbors. We apply this process to both CNN outputs, but then process the confidence maps and the selected keypoints differently. The cow keypoints are separated into upper body region keypoints, and leg and hoof region keypoints. If one upper region keypoint is detected by both CNNs, then we compute the final keypoint position to be the average of the two positions. But if this occurs for a leg and hoof region keypoint, we use the position identified by the color CNN as final output; this is because the leg and hoof movements are difficult to observe within frame difference images.

The output of this step are lists of body-part candidates. Formally, for a given frame at time t, all these body-part candidates can be represented as $p_j^{i,t} = (x, y)$, where j is the index of that body part, and $i \in \{1, 2...\}$ indicates the count of all possible keypoints extracted for this body part. The total number of i is not determined because the number of cow objects is unknown at this stage, and there could be some incorrectly-detected candidates. All these candidates are further selected and clustered in the next step.

3.4.2. Spatial clustering

The second step in the post-processing module is spatial clustering.

This step selects the correct body parts and clusters them into different cow objects. The first task before clustering is to determine the number of cows in the frame by counting cow centers. The cows in the frame will not overlap with each other because of the fences, so the number of cows can be estimated based on the keypoint clusters. Given a set of extracted keypoint candidates $p_j^{i,t}$ from the upper body parts, the corresponding cow center positions can be estimated based on the constraints of the keypoints, shown in Eq. (2).

$$c_j^{i,t} = F_j^{-1}(p_j^{i,t}) \tag{2}$$

Then a mean-shift clustering method is applied to the 2D spatial positions of all the cow centers $c_j^{i,t}$. Because the mean-shift algorithm is a non-parametric clustering technique, we only need to specify the minimum distance between two neighboring clusters; the number of clusters is not pre-defined. The minimum distance should be set to be much smaller than a cow's width, which is 100 pixel in our experiment. Based on the clustering results of the center points, the corresponding body parts are labelled into separate cow objects. We accept a cow object if the system detects more than half of its keypoints, which is a sufficient amount to localize the closed polygon of the body region.

The cow centers are also used to predict the location of missing body parts that the network fails to detect. After all keypoints are clustered into distinct cow objects, then for each cow object, we compute the averaged cow centers based on the detected points, and the miss-detected keypoints can be estimated using the keypoint constraints F_j. While the location of these body parts are only estimates, they provide sufficient information of the cow's spatial location to be useful when searching for keypoints in leg-hoof regions.

The final process in this step is to match the leg-hoof points. Similar to Zhao et al. (2018), we indicate the region of all possible leg-hoof points using a rectangle that is one-third wider than the rectangle of the upper body. Candidates outside this region will not be considered. The search process relies on the structural model. We follow the order of *shoulder/tailhead*, *leg*, *hoof* along each limb, and search the joints from among the candidates that lie in the search range. We also reject inappropriate points by applying the rule that each limb should have a certain rotation range; the angle between *shoulder* to *leg* and *leg* to *hoof* must be greater than 90 degrees for valid keypoints. Finally, all the selected leg-hoof joints are connected to the body contour to complete the final cow structural model.

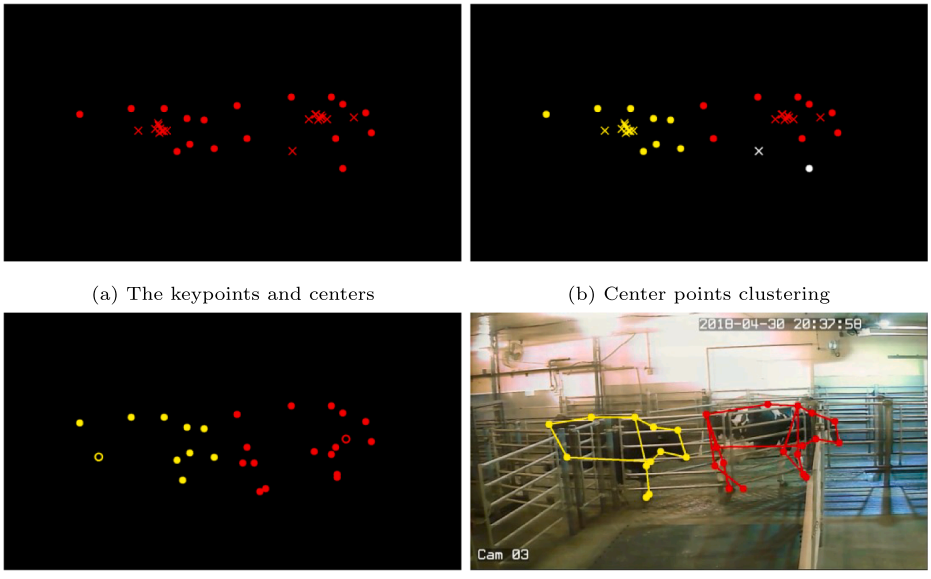
Fig. 4 illustrates the procedures of the spatial clustering step. The top left image shows the original extracted body parts from the previous step. The red circles are the extracted candidates and each is converted to a corresponding cow center, shown as crosses. Then in the top right image, all center crosses are clustered using mean-shift to produce three clusters shown in distinct colors. Here the incorrect cluster (white) is eliminated because there are not enough candidates. Next in the bottom left image, points in the leg and hoof region are assigned to each cow object. The empty circles are predicted points; the yellow one is blocked by the fences. Finally, by connecting all keypoints together, we form two cow structural models as shown in the bottom right image.

3.4.3. Temporal filtering

The final step in post-processing module refines the detection results using temporal information and matches cow objects across different frames. The two previous steps each operate on a single image, but the relationship between neighboring video frames is helpful to refine keypoint positions. It is reasonable to assume that the cows walk on an identical path between the fences and that they move steadily and slowly. This means that for a specific keypoint in the upper body, its trajectory over time should be smooth and any points far from the trajectory line can be considered outliers.

Based on this idea, we refine the positions of every upper body-part point across time to improve the temporal smoothness of the output. Before this step starts, all the frames in a video have been processed, so we know the number of cow objects in each frame. Then for every body part in the upper body region, we temporally filter each trajectory to remove and correct the outliers. In our experiment, we use a median filter, which is simple and provides robust prediction. Other filters such as the Kalman filter do not work well especially when there are too many missing points from the previous steps. The leg-hoof region points are not involved in this process, since their trajectories are much more complicated.

Based on the trajectories of each cow object, the cow objects can be matched between neighboring frames. After this process, the system detects the total number of cows shown in a complete video sequence, and parameters about how every cow moves can be inferred, including the speed and rhythm or the walking poses (Whay, 2002).



(a) The keypoints and centers (b) Center points clustering

(c) Adding leg region points (d) The output structural model

Fig. 4. The procedure of spatial clustering during post-processing. Circles represent the body parts p and crosses are the estimated cow centers c. Empty circles are the predicted body parts. Each color indicates a different cow object.

4. Evaluation metrics

This section introduces our evaluation metrics. Although our method uses few ground-truth labels for training, ground truth is typically also required for performance evaluation. Therefore, in this study we propose to use both supervised measures, which compare the detected results with ground-truth labels (i.e. manually labelled points without noise or obstacles), and unsupervised measures, which directly evaluate the results without labels. Adding unsupervised measures to the evaluation process improves its thoroughness in the presence of insufficient labels. We first discuss the supervised measures for the cow structural model, and then introduce two unsupervised metrics.

4.1. Supervised measures

Quantifying the performance of the cow structural model requires more than the typical measures used to quantify object detection. As mentioned in Section 1, the cow structural model is designed to provide two types of information: the spatial location of the body region, and the detailed positions of body parts. Both information is represented in terms of the keypoints of the cow body parts, and our ground-truth labels are also in terms of keypoints. As a result, we separately evaluate the area of the cow body region and the points in the leg-hoof region. Two metrics are developed and described below in detail: the *Body F1 score* and the *Leg-hoof F1 score*. In each case, the F1 score is harmonic mean of precision and recall when comparing the detection results to the ground truth. Both metrics compare accuracy at the keypoint level. In a later experiment in Section 5.4, we also propose a method to convert the cow structural model to a binary mask with both body region and extended limbs, for the sole purpose of comparing our detected keypoint model with other mask-based segmentation methods.

4.1.1. Body F1

This metric measures the spatial area formed by the body region points. We connect the keypoints in the upper body region and generate one polygon mask for both the detected structural models and the ground-truth keypoints. Then we compare the two masks using the typical Intersection Over Union (IOU) metric and report the F1 score.

4.1.2. Leg-hoof F1

For the legs and hooves, a single pixel position represents each keypoint. However, physical joints typically extend for a larger spatial region. Therefore, the evaluation metric must accommodate this discrepancy, which may introduce systematic errors to both the labelling and detection process. For this reason, when measuring the distance between ground truth and the detected leg and hoof keypoints, we set a threshold distance of 30 pixels, which is the minimum resolution of the DeepLabCut labelling system. If the distance between the points is less than this threshold, we consider the joint to be detected, and points further away are considered to be miss-detected. After thresholding, we determine how many leg-hoof points are successfully detected, and summarize this using the F1 score computed from the precision and recall. Since we do not create ground-truth labels for keypoints that are completed blocked by obstacles, these blocked joints do not affect the evaluation result.

4.2. Unsupervised measures

Unsupervised measures allow performance evaluation without ground-truth labels. This is particularly critical for video, where exhaustive application-specific labeling becomes even more onerous. Without labels, previously proposed metrics such as mean of region similarity, contour accuracy (Li et al., 2017), and temporal stability metric (Perazzi et al., 2016) cannot be computed. Here, we apply prior knowledge to evaluate the performance when the ground-truth labels are not provided.

We consider two rules for the cow structural model. First, the spatial locations of the keypoints in a model should always form a cow-shaped object. Second, the shape of the cow body should be stable during the walk and the keypoints should have similar smooth trajectories. Based on these two constraints, we introduce two unsupervised metrics: the valid cow percentage and temporal consistency.

4.2.1. The Valid Cow Percentage (VCP)

This metric counts the fraction of detected cow models that are valid. Here valid means that the positions of the keypoints in the structural model can form a cow-shaped object. Like the supervised measure, we validate the upper body region and leg-hoof region separately.

For the upper body region, we use the trained keypoint constraints (Fig. 2) as a reference, and compute the similarity between the detected contour and the reference using the Fréchet distance (Alt and Godau, 1995). We choose this distance because it better captures the similarity between two curves, which are the body contour in our case. The computed distance is thresholded to form a binary decision whether the upper body region is valid or not. This distance threshold is set to 30 in our experiments. For points in the leg-hoof region, we define two interpretable rules to validate their spatial positions: all leg-hoof points should be lower than the body region points, and all hoof points should be lower than their corresponding leg points. If all leg-hoof points satisfy these two rules and the upper body region contour is also validated, the cow structure is considered valid.

This validation scheme is applied to all the detected cow objects in a video sequence, and the Valid Cow Percentage (VCP) is computed as the number of valid cow objects divided by the number of detected cows. The absolute VCP score is directly related to the actual number of cows in the testing video sequence, so the score is only meaningful when compared with other methods on the same testing dataset.

4.2.2. Temporal Consistency (TC)

The second unsupervised metric evaluates the Temporal Consistency (TC), which reflects the smoothness of the motion of moving objects in a video sequence. It is reasonable to assume that at a certain camera angle, the points from the body region always share the same translational motion because the shape of the cow body is stable. So ideally, the motion vector between every upper body region keypoint generated from one frame to the next frame should be the same. The Temporal Consistency (TC) metric evaluates this co-movement and computes the distance between the motion vectors generated by the body parts, which shows the smoothness of the detected keypoints in temporal domain.

Formally, for each keypoint in the upper body region p_j^t in a cow object, we compute its motion vector from time t to $t+1$ and summarize the variations d between all the motion vectors as

$$d^t = std(p_j^{t+1} - p_j^t), \forall\, j \in \{j_1, j_2 ...\} \tag{3}$$

where *std* is the standard deviation, and j_i represents the index of the body part from the upper body region. Then the temporal consistency is computed as the average of the motion vector distance across all the frames in a video sequence.

$$TC = mean(d^t), \forall\, t \tag{4}$$

This measure is applied to every individual cow object in a video sequence, and smaller TC values imply smoother object movements. The variation of the TC value is not considered because the cows are walking slowly and steadily in the path, there are unlikely to be large TC variations

Table 1
Summary of three sets of video data used in the experiments. The # Pixel per cow is in units of millions.

	Set 1	Set 2	Set 3
Capture Device	DVR	GoPro	IP camera
Video info	1280∗720 @12fps	1232∗384 @30fps	1920∗1088 @30fps
# Pixels per cow	0.88 m	0.29 m	1.35 m
Image Quality	Low	Low	High
Field of view	Narrow	Wide	Wide
# cow per clips	Single	Multiple	Multiple
# video clips (# for training)	87 (5)	18 (2)	114 (5)
# training frames	100	40	100
# testing frames	585	59	611

5. Experiments

5.1. Data collection

All cow videos in these experiments are collected from the Purdue Animal Sciences Research and Education Center located in West Lafayette, IN, USA from 2018 to 2019. All procedures were approved by the Institutional Animal Care and Use Committee (PACUC #1803001704). The cameras are mounted at fixed positions include a side-view of the pathway where cows walk every day. This path has fences on both sides and only allows one cow to walk through at a time. This limits the amount of cow-overlap; however the dense fences partly block the view of the cows, and some body parts are not visible behind the fences, as shown in Fig. 1. This walking path is a typical component of many dairy farms.

During the course of data collection, we used three different capture devices: a commercial surveillance camera with Digital Video Recorder (DVR), a GoPro camera, and a high-quality IP camera. Table 1 shows the detailed information of the three video sets captured from the three cameras. The DVR videos have the worst quality with low frame rate and low resolution. The GoPro videos provide higher frame rate, but they are spatially cropped with less spatial details. The IP camera captures high quality videos with both high frame rate and rich spatial information.

Table 1 compares several factors among the cameras that will influence detection performance. As noted, the video resolution and frame rate are different between the three sets, and Set 3 has the best quality. The number of pixels per cow refers to the average number of pixels that each cow occupies in an image, which is an indication of the spatial detail in each set. The Set 2 videos only have 0.29 million pixels per cow, which is less than a third of the other two sets. The field-of-view for each camera is also different. Set 1 videos only capture the center of the walking path where there are fewer fences, while the other two sets capture a wider view which includes two sides that have denser fences. In addition, the typical number of cows in one video are different across the sets. Narrow field-of-view videos normally capture a single cow in the frame, but the wider-angle videos could contain multiple cows, which challenges the detection method. In general, Set 3 has more video clips than the others with the greatest variety, so we will further divide this set into subsets in a later experiment described in Section 5.4.

To prepare the videos, we temporally segment the hours-long sequences into 10-s clips, on average, where all cows walk from left to right. In each set, we separate the clips into training and testing groups, where the number of training clips per set are shown in parentheses in Table 1 after the number of video clips. All multiple-cow clips are testing clips, so the training clips all contain only a single cow object. Non-consecutive frames are chosen randomly for labeling from both training and testing clips as described in Section 3.3.

5.2. System component evaluation

This experiment compares all the internal outputs from our proposed system shown in Fig. 3, to demonstrate the importance of each individual module. We choose the output of CNN1 as the baseline method, which is the original method in the DeepLabCut (DLC) toolbox (Mathis et al., 2018). However, this method can only detect one object per frame, so for a fair comparison, we only use the videos in Set 1 since these only contain one cow object. We compare the baseline method with four other internal outputs from the system: the CNN2 output from the difference videos, the CNN1 output plus the Post-Processing (PP) stages, the CNN2 output plus the PP stages, and the final merged result.

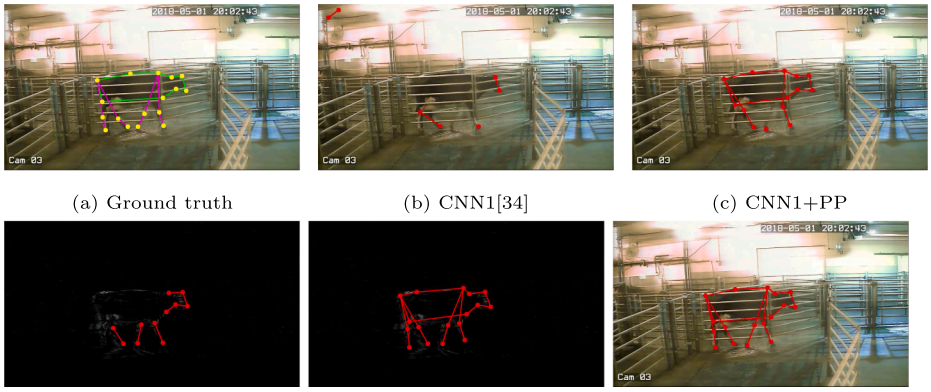
The implementation details are explained below. The frame difference images are generated by the sum of differences between the current frame and both the previous and next frame. The training labels from Set 1 are used to fine-tune both CNNs in the system. Recall that CNN1 processes the color images and CNN2 processes the frame-difference images. Both networks are pre-trained on ImageNet (Krizhevsky et al., 2012) and their final upsampling layers are fine-tuned with our cow images. For the two CNN methods without PP stage, we follow the extraction method from the DeepLabCut toolbox by setting a hard threshold and finding the location in the confidence maps with the maximum probability.

Both supervised and unsupervised evaluation metrics are used, but their testing data are different. For unsupervised measures, we compare the Valid Cow Percentage (VCP) and Temporal Consistency (TC) for all the frames in the testing videos because no labels are required. But for supervised measures, only the 585 labelled testing frames are used for evaluation. Among these labelled images, we report the body F1 score and leg-hoof F1 score, and the VCP score is also computed to compare the cow detection capability of each module in the system. Both qualitative and quantitative results are presented below.

Fig. 5 shows an example of all five outputs of one testing image in Set 1. The direct outputs from the two CNNs without post-processing (top middle and bottom left) miss-detect some body parts, because they apply the strategy from the original DLC method that only selects one maximum point. Our proposed post-processing module uses non-maximum suppression to select all local maximum values from the confidence map, and all body-part candidates are detected (see bottom right of Fig. 5). Considering the leg-hoof points, some joints of the swing leg are missed by CNN1 based on color image, because of motion blurriness and heavy compression. But these points are detected by CNN2 using the frame difference image, and the merged result generates a complete cow structural model.

The numerical comparison results are presented in Table 2. In general, our complete system (last row) improves the performance compared to the method in the DLC toolbox (first row). It can be observed that adding a Post-Processing (PP) module largely improves the system performance. The temporal and spatial prediction in the PP module improves the cow-detection ability demonstrated by the increasing VCP scores. The two VCP scores from supervised measure and unsupervised measures are not comparable because their test sets are different. In addition, the temporal filtering process in the PP module largely improves the Temporal Consistency (TC), because the original CNN method purely operates on an image without considering temporal information. Comparing two F1 scores in the supervised measures, the PP step improves the detection accuracy for the cow structural model because more body-part candidates are selected from the intermediate CNN output.

Comparing the first two rows from the table, we can see CNN2 has better performance than CNN1 for the cow body region but works poorly on the leg and hoof regions. As explained in Section 3.3, CNN2 operates on gray-scale edges generated by the frame difference and better captures smoothly moving objects like the body region. But it cannot work in isolation because it eliminates too much information contained in the original images, such as the stationary legs. As a result,



(a) Ground truth (b) CNN1[34] (c) CNN1+PP

(d) CNN2 (e) CNN2+PP (f) Merged output

Fig. 5. The outputs of different stages in the proposed system.

Table 2
Comparison of the outputs of the system components on Set 1 videos (single-cow). Smaller TC value means smoother object movement in the video. Bold numbers show the best performance method in each column.

	Unsupervised		Supervised		
	VCP	TC	VCP	Body F1	Leg-hoof F1
CNN1 (DLC(Mathis et al., 2018))	0.447	102.8	0.714	0.260	0.391
CNN2	0.408	155.0	0.673	0.366	0.252
CNN1 + PP	0.632	**8.9**	0.846	0.772	0.373
CNN2 + PP	0.667	10.2	0.929	0.841	0.333
Merged output	**0.705**	9.0	**0.960**	**0.879**	**0.434**

merging the two networks together obtains better detection for the leg-hoof region points.

5.3. Dataset robustness evaluation

This experiment evaluates the system robustness with different datasets. Training-based detection methods normally perform worse when they are applied to testing data that is substantially different from the training set. In this experiment, we evaluate the performance of our system when testing on frames collected from the three different cameras, that capture the same region of the farm but with different capture angles. This experiment also explores the influence of image quality on our system, since the video qualities from the 3 sets are also different.

For the training images in each video set, we fine-tune three detection systems, $S1$, $S2$, and $S3$, based on each individual corresponding datasets, respectively. An extra system S_all is trained on all the training frames together. In the testing phase, each trained system is applied to the images from the three sets separately. We also test each system on all testing images together for an overall comparison. All training and testing data are separated regardless of their dataset, and no images used for both training and testing. In total, there are 4 trained models testing on 4 groups of test sets, which forms 16 training/testing pairs. For each pair, we measure the final system output using supervised metrics: body F1 score and leg-hoof F1 score. Table 3 shows the comparison results.

In Table 3, each row represents a system trained from one dataset, and each column shows the system performance on one corresponding test set. Comparing the four systems, it can be observed that S_all achieves similar and slightly better performance than the others, and this merged system even works better than when each individual system is both trained and tested on its own videos (diagonal values). This demonstrates that adding training data helps to improve the detection performance for video sets that were captured in the same environment but from slightly different angles.

Table 3
System performance comparison on different video sets. The bold numbers show the best performance of each column.

Trained system	Body F1 score on				Leg-hoof F1 score on			
	Set1	Set2	Set3	All	Set1	Set2	Set3	All
$S1$	0.80	0.42	0.51	0.64	0.61	0.18	0.35	0.46
$S2$	0.72	**0.65**	0.58	0.65	0.16	0.59	0.33	0.26
$S3$	**0.82**	0.56	0.59	0.69	0.61	0.52	**0.56**	0.58
S_all	**0.82**	0.64	**0.61**	**0.71**	**0.62**	**0.65**	**0.56**	**0.59**

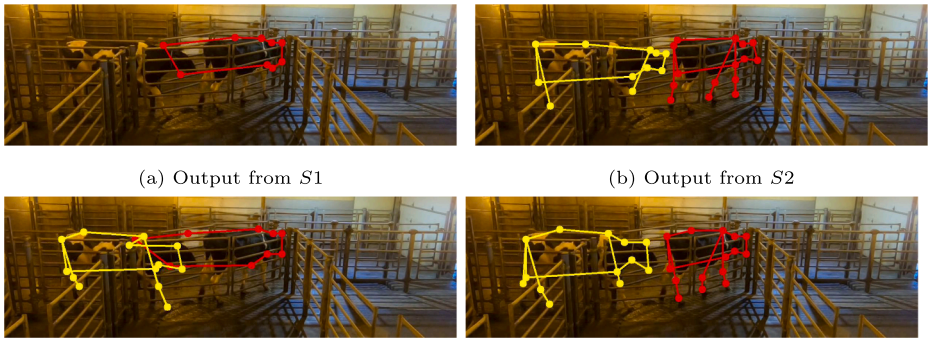
The results in Table 3 also allow us to examine the performance of the method when the input videos have different qualities. While both Set 1 and Set 2 have low quality, the images in Set 2 shown in Fig. 6 have a small spatial resolution while the images in Set 1 from Fig. 5 are blurry with poor illumination. Therefore, the results of system $S1$ on Set 2 and of system $S2$ on Set 1 images are poor, especially for the leg and hoof regions. However, system $S3$, which is trained on high quality images, provides better results on both these two datasets. This demonstrates that using higher quality images or increasing the variation of training data can improve system performance.

A final observation from the table is that the body region F1 scores are more stable across different systems than the leg-hoof F1 scores, due to the fact that the post-processing module that only operates on the body region. The spatial and temporal prediction in the post-processing model improve the estimation of missing and incorrectly detected points, which compensates for poor CNN performance. Since the legs and hooves are estimated directly from the CNN outputs, the performance variation is primarily due to the variation of training data.

In addition to numerical comparison, in Fig. 6 we also present some visual results from all 4 trained systems applied to a test image from Set 2 that contains two cows. Comparing the outputs, system $S1$ fails to detect two cow objects and $S3$ is confused with some body parts between the two cow objects. However, system $S2$ and S_all both detect two cow objects and present an accurate cow shape, because these two systems are both trained with data from Set 2. But the merged result from S_all is more accurate on some body parts, for example the points on each cow's back, because of the additional training data involved. However, for the leg and hoof region, none of the systems detect all the points, due to the difficulty of observing them and the lack of post-processing process.

5.4. Segmentation methods comparison

This experiment compares the detection performance between our system and other popular object detection methods. Recall that the



(a) Output from $S1$ (b) Output from $S2$

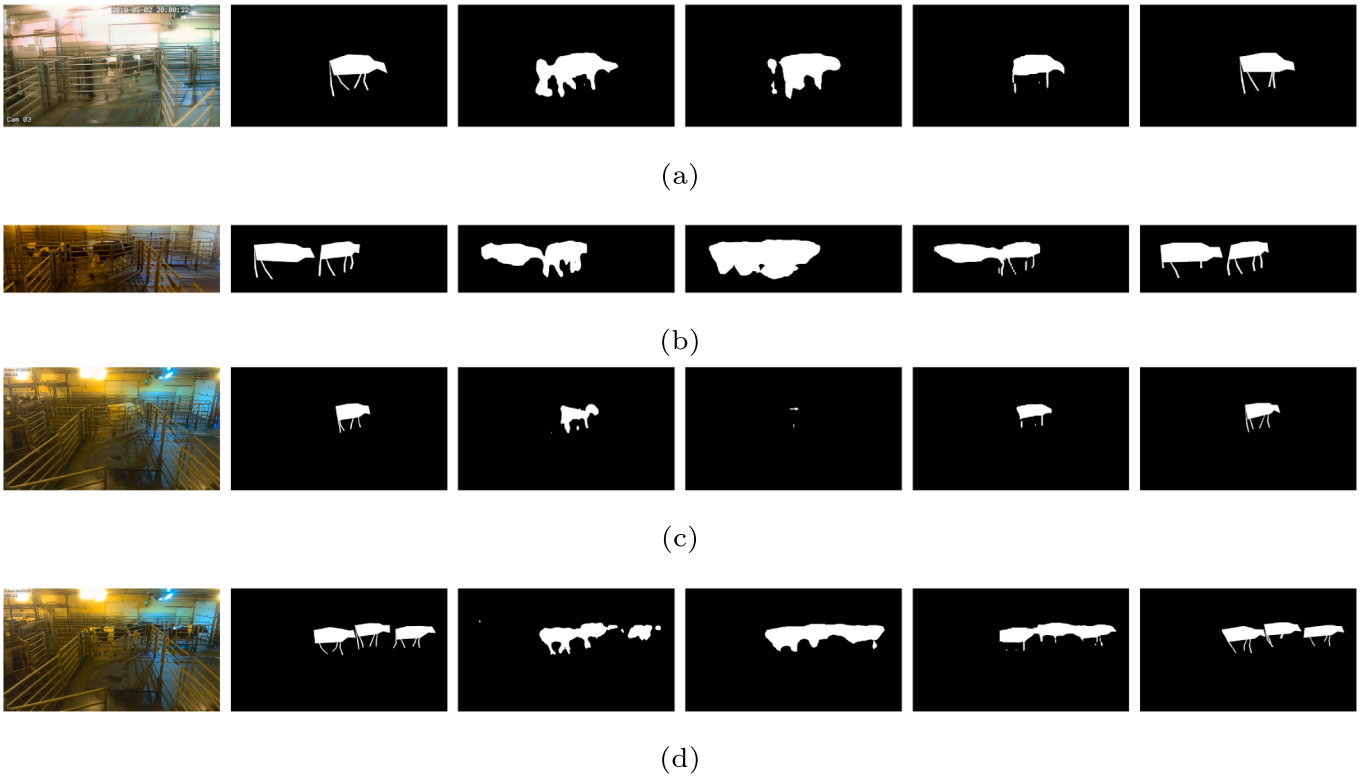
(c) Output from $S3$ (d) Output from S_all

Fig. 6. The detection comparison between systems trained on different video sets. This example image is from Set 2.

motivation for our system is not only to segment the spatial location of the cow, but also to detect critical keypoints about its body parts. Therefore, ideally comparison methods should also target these two goals. However, as mentioned in the introduction, most previous keypoint detection methods focus on human objects and incorporate knowledge about human body parts, and it is difficult to adapt them to cow bodies for a comparison. On the other hand, there are many popular object detection methods which can be fine-tuned to segment cows, and these make for an effective comparison. In this experiment, we compare the cow object detection performance between our system and other three popular pixel-wise object detection methods: One Shot Video Object Segmentation (OSVOS) (Caelles et al., 2017), DeepLab (Chen et al., 2017), and Mask R-CNN (He et al., 2017).

To create a performance comparison that does not disadvantage the object detection methods, we convert the output of our structural model into a binary cow mask, with two steps. First, all keypoints from the upper body region are connected to form a closed area representing the cow body. Second, every leg-hoof limb is expanded from a line into a polygon with a horizontal width of 20 pixels, as shown in the second column of Fig. 7. This expansion process is applied to both the ground-truth labels and the detection results. The newly expanded ground-truth masks are then used to fine-tune the object detection methods, as well as to compute performance metrics. Still the point-to-mask conversion is not perfect. The approximated masks cannot exactly cover the cow object from the original image; see for example the inaccurate edges of the cow body and the straight legs.

We use all the training and testing data from the three video sets in this experiment. In total, there are 240 single-cow frames for training and 1255 images for testing. Each of the three comparison methods are fine-tuned with the approximate cow masks, with different implementation details. For OSVOS (Caelles et al., 2017), we use the parent network pre-trained on the DAVIS 2016 (Perazzi et al., 2016) dataset and fine-tune it with our data. The output results are binarized using Otsu (Liao et al., 2001) threshold. For DeepLab (Chen et al., 2017), we use the pre-trained network from the COCO dataset (Lin et al., 2014), and we modify the last layer to produce two classes: cows



(a)

(b)

(c)

(d)

Fig. 7. Results using different detection methods. From left to right: original image, ground truth, OSVOS (Caelles et al., 2017), DeepLab (Chen et al., 2017), Mask R-CNN (He et al., 2017), and ours. Example (a) and (b) are from Set 1 and Set 2, respectively; example (c) and (d) are both from Set 3.

and background. The fine-tuning process is applied only on the last atrous spatial pyramid pooling layers with binary entropy loss. For Mask R-CNN (He et al., 2017), we use the network pre-trained on the COCO dataset and fine-tune its region proposal network and feature pyramid network. The classifier outputs are also adjusted to the two classes of cows and background.

Fig. 7 shows some visual examples of the detection results. From left to right are the original image, ground truth, and the results from OSVOS, DeepLab, Mask R-CNN, and our system, respectively. Each row shows an example which is selected from a different test set. Example (a) includes a human wearing black clothes who is walking right behind the cow. This confuses OSVOS which considers it to be part of the moving foreground object. Example (c) shows a special case which contains a pure white cow, and this color is not present in the training data. The DeepLab method completely misses the cow, because it directly extracts information from the color image and this rare color has not been seen before. The OSVOS method detects part of this cow using motion information, but Mask R-CNN works well because its region proposal network determines there is an object candidate and segments the cow object correctly.

Examples (b) and (d) contain multiple cows, and each method does detect multiple cow objects. However, the three masked-based methods merge all detected cow objects together because the objects are close to each other, and we need further effort to count the number of cows or to extract other detailed information. But our result provides a clear delineation between the cow objects, due to the use of the structural model. Another observation about these two examples is that the cow positions in these two images are different. Some cows are in the middle with fewer fences and others are on either the left or right side with denser fences blocking the view. Every method can detect the middle cow, but the cows on the sides are more challenging to detect due to the obstacles. We further analyze the influence of fences in later paragraphs.

Numerical comparison results among the methods are also reported using the F1 scores of the Intersection Over Union (IOU) between the detection results and the ground-truth masks. The measures are reported based on every test set separately in Table 4, and on distinct subsets of Set 3 in Table 5.

From Table 4, it can be observed that our method achieves the highest IOU score for most sets, although its performance relative to the fine-tuned Mask R-CNN is similar. There are three factors which may influence these IOU scores. First, when comparing the masks using IOU, we use a merged mask containing both the cow body and leg regions. Since the body region occupies a larger area of the ground-truth mask, the IOU score can still be high even if the legs are miss-detected. Second, because the masks for our method and the ground truth are both converted from keypoints, it is highly sensitive to the positions of the keypoints, especially for the narrow leg regions. Small position shifts can lead to a large change to the converted mask, which will influence the IOU score. Third, when our system does not detect a leg or hoof point, the mask will be empty in this region. This will also decrease the IOU of our system. Nonetheless, our system performs well in comparison.

As mentioned above, a main consideration of our system is to obtain acceptable performance even when there are multiple cows, and when there are obstacles like fences. We use Set 3 videos to further explore the influence of these issues, to eliminate any performance variations due to video quality. As Fig. 7 shows, Set 3 images have a wider view of the walking path, and cows in the center have fewer fences while cows on the left or right sides are blocked with denser fences. So we separate the testing frames from Set 3 into four subsets: cows in the middle, cows on either side, single-cow frames, and multiple-cow frames. Among the four subsets, images with cows in the middle and with a single cow set will be easier than images from the other two subsets. The qualitative comparison F1 scores of these subsets are shown in Table 5. From the table, Mask R-CNN has better performance on the easier test case when the cows are blocked by fewer fences. But for difficult test sets like denser obstacles, our proposed system works better. The OSVOS method also performs well when there are more obstacles because this method only considers the foreground and background, which allows it to separate the stationary fences from the moving cows.

In general, compared to the other three mask-based object detection methods, our proposed system has three advantages. First, based on the keypoints detection, our method can correctly detect the cow structure even when the cows are behind the fences or there are humans nearby. Second, when there are multiple cow objects, this system can explicitly isolate each cow even when they are close to each other. Third, it can detect cows with color patterns that do not exist in the training data through the use of frame difference images. However, our system also has two limitations. First, the cow structural model completely depends on the accuracy of the body parts, and one inaccurate detection can cause large errors for the body contour and influence the overall spatial location. Second, the prediction system in our method is based on the keypoint constraints from the cow structural model, which is fixed after the training process. If there are not enough cow body parts detected, the prediction system still forces the results to conform to a particular shape, which could cause incorrect results.

Table 4
Comparison of methods on different test sets in terms of Intersection Over Union (IOU) scores ranging from 0 to 1. Larger score means larger overlapping regions, which means better performance.

	Set 1	Set 2	Set 3	All
OSVOS (Perazzi et al., 2016)	0.571	0.580	0.570	0.571
DeepLab (Chen et al., 2017)	0.655	0.513	0.577	0.610
Mask R-CNN (He et al., 2017)	0.735	**0.692**	0.630	0.682
ours	**0.750**	0.668	**0.662**	**0.703**

Table 5
Comparison of methods on subsets of Set 3 in terms of Intersection Of Union (IOU) scores. *Middle* means the cow is in the image center which has fewer obstacles, while *Side* means the cows are on the two sides with denser fences.

	Middle	Side	Single-cow	Multiple-cows
OSVOS (Perazzi et al., 2016)	0.672	0.589	0.650	0.547
DeepLab (Chen et al., 2017)	0.644	0.537	0.616	0.518
Mask R-CNN (He et al., 2017)	**0.749**	0.574	0.703	0.520
ours	0.734	**0.645**	**0.711**	**0.587**

6. Conclusions and Future Work

In this work, we design a practical system to detect the structural information for cows recorded in video. We use keypoints to form a cow structural model, which represents both the cow's overall spatial location and the positions of its specific body parts, such as the joints from the leg and hoof. The proposed detection system applies two CNNs to extract the keypoints from raw images, and a post-processing model is developed to select individual points and convert them into cow structural models. This system can detect and track multiple cow objects at the same time, and it is also effective when applied to different quality videos that have been captured on commercial farms during normal operation.

In future work, we will apply this system to address several potential applications. Potential applications include visual cow weight estimation, cow re-identification, and even lameness analysis based on the head movement and back arch (Poursaberi et al., 2010). In addition, we will explore new methods to improve the limitations of our current system. In particular, we will improve the robustness of the leg and hoof keypoint estimation. These keypoints are challenging to detect due to the motion blur and poor lighting. However, the trajectory of these points is important for the cow lameness detection process. Therefore,

designing new methods to overcome these challenges will improve the applicability of our current system. Finally, dairy cows are just one type of four-legged livestock, and we anticipate our method can be easily extended to accommodate similar animals.

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.

References

Alt, H., Godau, M., 1995. Computing the Fréchet distance between two polygonal curves. Int. J. Comput. Geometry Appl. 5 (01n02), 75–91.

Andrew, W., Greatwood, C., Burghardt, T., 2017. Visual localisation and individual identification of Holstein Friesian cattle via deep learning. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2850–2859.

Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B., 2014. 2D human pose estimation: New benchmark and state of the art analysis. In: IEEE Conference on Computer Vision and Pattern Recognition.

Aujay, G., Hétroy, F., Lazarus, F., Depraz, C., 2007. Harmonic skeleton for realistic character animation. In: Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation, Eurographics Association, pp. 151–160.

Caelles, S., Maninis, K.-K., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L., 2017. One-shot video object segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 221–230.

Cao, Z., Simon, T., Wei, S.-E., Sheikh, Y., 2017. Realtime multi-person 2D pose estimation using part affinity fields, in. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7291–7299.

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., Sheikh, Y., 2018. OpenPose: realtime multi-person 2D pose estimation using part affinity fields, arXiv preprint arXiv:1812.08008.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2017. DeepLab: Semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs. IEEE Trans. Pattern Anal. Mach. Intell. 40 (4), 834–848.

Cheng, J., Tsai, Y.-H., Wang, S., Yang, M.-H., 2017. SegFlow: Joint learning for video object segmentation and optical flow, in. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 686–695.

Condotta, I.C., Brown-Brandl, T.M., Stinn, J.P., Rohrer, G.A., Davis, J.D., Silva-Miranda, K.O., 2018. Dimensions of the modern pig. Trans. ASABE 61 (5), 1729–1739.

Cook, N., 2020. Symposium review: the impact of management and facilities on cow culling rates. J. Dairy Sci. 103 (4), 3846–3855.

Cook, N.B., Nordlund, K.V., 2009. The influence of the environment on dairy cow behavior, claw health and herd lameness dynamics. Vet. J. 179 (3), 360–369.

Fleishman, L., Endler, J., 2000. Some comments on visual perception and the use of video playback in animal behavior studies. Acta ethologica 3 (1), 15–27.

Günel, S., Rhodin, H., Morales, D., Campagnolo, J., Ramdya, P., Fua, P., 2019. Deepfly3D: A deep learning-based approach for 3D limb and appendage tracking in tethered, adult drosophila. bioRxiv, 2019 640375.

Guzhva, O., Ardö, H., Herlin, A., Nilsson, M., Åström, K., Bergsten, C., 2016. 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. Comput. Electron. Agric. 127, 506–509.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

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.

Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., Schiele, B., 2016. DeeperCut: A deeper, stronger, and faster multi-person pose estimation model. In: European Conference on Computer Vision. Springer, pp. 34–50.

Kawakatsu, T., Kakitani, A., Aihara, K., Takasu, A., Adachi, J., 2017. Traffic surveillance system for bridge vibration analysis. In: 2017 IEEE International Conference on Information Reuse and Integration (IRI). IEEE, pp. 69–74.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105.

Lee, Y.J., Kim, J., Grauman, K., 2011. Key-segments for video object segmentation. In: IEEE International Conference on Computer Vision, pp. 1995–2002.

Leonard, S.M., Xin, H., Brown-Brandl, T.M., Ramirez, B.C., 2019. Development and application of an image acquisition system for characterizing sow behaviors in farrowing stalls. Comput. Electron. Agric. 163, 104866.

Liao, P.-S., Chen, T.-S., Chung, P.-C., et al., 2001. A fast algorithm for multilevel thresholding. J. Inf. Sci. Eng. 17 (5), 713–727.

Li, X., Qi, Y., Wang, Z., Chen, K., Liu, Z., Shi, J., Luo, P., Tang, X., Loy, C.C., 2017. Video object segmentation with re-identification. arXiv preprint arXiv:1708.00197.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., Microsoft, C.O.C.O., 2014. Common objects in context. In: European Conference on Computer Vision. Springer, pp. 740–755.

Maninis, K.-K., Caelles, S., Chen, Y., Pont-Tuset, J., Leal-Taixé, L., Cremers, D., Van Gool, L., 2018. Video object segmentation without temporal information. IEEE Trans. Pattern Anal. Mach. Intell. 41 (6), 1515–1530.

Mathis, A., Mamidanna, P., Cury, K.M., Abe, T., Murthy, V.N., Mathis, M.W., Bethge, M., 2018. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. Nat. Neurosci. 21 (9), 1281–1289.

Neubeck, A., Van Gool, L., 2006. Efficient non-maximum suppression. In: 18th International Conference on Pattern Recognition (ICPR'06), vol. 3, pp. 850–855.

Newell, A., Yang, K., Deng, J., 2016. Stacked hourglass networks for human pose estimation. In: European Conference on Computer Vision, Springer, pp. 483–499.

Perazzi, F., Pont-Tuset, J., McWilliams, B., Van Gool, L., Gross, M., Sorkine-Hornung, A., 2016. A benchmark dataset and evaluation methodology for video object segmentation, in. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 724–732.

Pereira, T.D., Aldarondo, D.E., Willmore, L., Kislin, M., Wang, S.S.-H., Murthy, M., Shaevitz, J.W., 2019. Fast animal pose estimation using deep neural networks. Nat. Meth. 16 (1), 117.

Pluk, A., Bahr, C., Leroy, T., Poursaberi, A., Song, X., Vranken, E., Maertens, W., Van Nuffel, A., Berckmans, D., 2010. Evaluation of step overlap as an automatic measure in dairy cow locomotion. Trans. ASABE 53 (4), 1305–1312.

Poursaberi, A., Bahr, C., Pluk, A., Van Nuffel, A., Berckmans, D., 2010. Real-time automatic lameness detection based on back posture extraction in dairy cattle: shape analysis of cow with image processing techniques. Comput. Electron. Agric. 74 (1), 110–119.

Redmon, J., Farhadi, A., 2018. YOLO v3: An incremental improvement, arXiv preprint arXiv:1804.02767.

Shao, W., Kawakami, R., Yoshihashi, R., You, S., Kawase, H., Naemura, T., 2019. Cattle detection and counting in UAV images based on convolutional neural networks. Int. J. Remote Sens. 1–22.

Song, X., Leroy, T., Vranken, E., Maertens, W., Sonck, B., Berckmans, D., 2008. Automatic detection of lameness in dairy cattle—vision-based trackway analysis in cow's locomotion. Comput. Electron. Agric. 64 (1), 39–44.

Song, X., Bokkers, E., van der Tol, P., Koerkamp, P.G., van Mourik, S., 2018. Automated body weight prediction of dairy cows using 3-dimensional vision. J. Dairy Sci. 101 (5), 4448–4459.

Spoliansky, R., Edan, Y., Parmet, Y., Halachmi, I., 2016. Development of automatic body condition scoring using a low-cost 3-dimensional Kinect camera. J. Dairy Sci. 99 (9), 7714–7725.

Ter-Sarkisov, A., Ross, R., Kelleher, J., 2017. Bootstrapping labelled dataset construction for cow tracking and behavior analysis. In: 2017 14th Conference on Computer and Robot Vision (CRV). IEEE, pp. 277–284.

Tokmakov, P., Alahari, K., Schmid, C., 2017. Learning video object segmentation with visual memory, in. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 4481–4490.

Toshev, A., Szegedy, C., 2014. Deeppose: Human pose estimation via deep neural networks, in. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1653–1660.

Tsai, Y.-H., Yang, M.-H., Black, M.J., 2016. Video segmentation via object flow. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3899–3908.

Viazzi, S., Bahr, C., Schlageter-Tello, A., Van Hertem, T., Romanini, C., Pluk, A., Halachmi, I., Lokhorst, C., Berckmans, D., 2013. Analysis of individual classification of lameness using automatic measurement of back posture in dairy cattle. J. Dairy Sci. 96 (1), 257–266.

Voigtlaender, P., Leibe, B., 2017. Online adaptation of convolutional neural networks for video object segmentation, arXiv preprint arXiv:1706.09364.

Wang, W., Shen, J., Porikli, F., 2015. Saliency-aware geodesic video object segmentation, in. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3395–3402.

Whay, H., 2002. Locomotion scoring and lameness detection in dairy cattle. Practice 24 (8), 444.

Zhao, K., Bewley, J., He, D., Jin, X., 2018. Automatic lameness detection in dairy cattle based on leg swing analysis with an image processing technique. Comput. Electron. Agric. 148, 226–236.

Zhao, K., Jin, X., Ji, J., Wang, J., Ma, H., Zhu, X., 2019. Individual identification of Holstein dairy cows based on detecting and matching feature points in body images. Biosyst. Eng. 181, 128–139.