Voxelized 3D Feature Aggregation for Multiview Detection

Jiahao Ma  
Australian National University  
Canberra, ACT, Australia  
jiahao.ma@anu.edu.au

Jinguang Tong  
Australian National University  
Canberra, ACT, Australia  
jinguang.tong@anu.edu.au

Shan Wang  
Australian National University  
Canberra, ACT, Australia  
shan.wang@anu.edu.au

Zicheng Duan  
University of Adelaide  
Adelaide, SA, Australia  
zicheng.duan@adelaide.edu.au

Chuong Nguyen  
CSIRO Data61  
Canberra, ACT, Australia  
chuong.nguyen@data61.csiro.au

Abstract—Multi-view detection incorporates multiple camera views to alleviate occlusion in crowded scenes, where the state-of-the-art approaches adopt homography transformations to project multi-view features to the ground plane. However, we find that these 2D transformations do not take into account the object’s height, and with this neglect features along the vertical direction of same object are likely not projected onto the same ground plane point, leading to impure ground-plane features. To solve this problem, we propose VFA, voxelized 3D feature aggregation, for feature transformation and aggregation in multi-view detection. Specifically, we voxelize the 3D space, project the voxels onto each camera view, and associate 2D features with these projected voxels. This allows us to identify and then aggregate 2D features along the same vertical line, alleviating projection distortions to a large extent. Additionally, because different kinds of objects (human vs. cattle) have different shapes on the ground plane, we introduce the oriented Gaussian encoding to match such shapes, leading to increased accuracy and efficiency. We perform experiments on multi-view 2D detection and multi-view 3D detection problems. Results on four datasets (including a newly introduced MultiviewC dataset) show that our system is very competitive compared with the state-of-the-art approaches. The code and dataset are publicly available at https://github.com/Jiahao-Ma/VFA.

Index Terms—Multi-view detection, synthesis dataset.

I. INTRODUCTION

Multi-view 3D object detection [2], [3], [9] on synchronized frames from different calibrated cameras is considered as an effective solution to the occlusion problem faced by a single camera. A multi-view perception system leverages images that come from different perspective views with overlapping fields of view to complement each other to alleviate blind spots in the field of view. The essence of multi-view detection is how to effectively fuse features from multiple camera views.

State-of-the-art systems [1], [10], [11] in this field leverage camera poses and are typically based on 2D transformations, i.e., projecting multi-view image feature onto ground plane. With such transformations the feature projection procedure follows the geometric constraints of the cameras, so that the same point on the ground plane can only aggregate features depicting the same object from multiple cameras.

However, these 2D transformations do not consider the 3D scene geometry, and would make incorrect projections of features along the Z axis (vertical direction) of same object. That is to say, features of the same object at different height levels are not projected onto the same position of the ground plane. In effect, while a position on the ground plane is supposed to aggregate features of the same object at that position, it actually receives features from a mixture of objects, leading to the polluted features. Such impure features not only compromise the system accuracy, but also slow down the training process. In Fig. 1, we showcase an example where the homography 2D transformation mistakenly projects four
colored points to different positions on the ground plane. Therefore, it is critical that our system is aware of such distortion to identify features that are on the same vertical direction in a multi-view system.

To address the problem, we propose voxelized 3D feature aggregation, or VFA, a 3D-aware projection method for multi-view object detection. In a nutshell, VFA explicitly considers the 3D clues by partitioning the 3D volum-of-interest into cubic voxels. This voxelization process allows us to identify the \( Z \) axis in the 2D multi-view images, so that we can identify and aggregate features on the same vertical line. Our method has the following characteristics and advantages. First, different from existing 2D transformation approaches, our method allows each position on the ground plane to receive and aggregate 2D features that are of the same vertical direction, thus mostly depicting the same object. This endows the aggregated features with a pure pattern, thus improving both recognition accuracy and training efficiency. Second, the underlying prior knowledge of our method is that objects are usually bounded in a vertical volume, which follows the force of gravity. As such, our method in principle is applicable to various objects such as human and cattle verified in this work.

In addition, to deal with objects (e.g., cattle) that have an oriented shape after being projected to the ground plane, we propose a simple technique applying oriented Gaussian distribution (OGD) to encode the ground truth location and orientation on the ground plane. This minor contribution brings further improvement in detection accuracy and training efficiency for multi-view cattle detection.

We perform extensive experiments on four datasets intended for multi-view detection, where we show our system yields very competitive accuracy compared with the state-of-the-art methods. Among the four datasets is a synthetic dataset, MultiviewC, newly introduced by this paper. The points made in this paper are summarized below.

- Major: an approach VFA to allow \( Z \)-axis-aware feature extraction aggregation for multi-view detection.
- Minor: an improved label embedding method, oriented Gaussian distribution, for directional objects (e.g., cattle) detection.
- A synthetic 3D multi-view dataset, MultiviewC, which includes cattle targets of diverse sizes and actions.

II. RELATED WORKS

Monocular object detection. Remarkable achievements have been made in 3D object detection over recent years. Due to the lack of depth information in the images, researchers have proposed fusion-based methods [15]–[18]. For instance, MV3D [15] encodes the sparse point cloud with multi-view representation and performs region-based feature fusion. Pointfusion [16] uses two networks to process images and raw point cloud data and fuses them at the feature level. In addition to predicting depth by fusing point cloud, increasing approaches [43]–[45] have been proposed to directly predict 3D Boxes using a convolutional neural network. Some methods [19], [20] are prone to perform 2D/3D matching via exhaustively sampling and scoring 3D proposals as representative templates. And some methods [8], [21], [22], [46] start with accurate 2D bounding boxes directly to roughly estimate 3D pose from geometric properties obtained by empirical observation.

Multiple views based detection. To detect objects under heavy occlusion, multiple perspective-based methods have been developed in the past few years. Fleuret et al. [9] proposes a probabilistic occupancy map to estimate the probabilities of occupancy using Multi-view streams jointly. In order to aggregate spatial neighbor features, Conditional Random Field (CRF) [6], [23] are exploited. Hou et al. [1] adopts feature perspective transformation to aggregate multi-view data and regresses pedestrian occupancy as a Gaussian distribution. Lima et al. [24] presents a multi-camera detection model without any training that tends to integrate multiple views via the graph-based method. Song et al. [11] proposes SHOT to alleviate the projection errors by multi-height-level homography transformation.

Inference in the bird’s-eye-view frame. Models using intrinsics and extrinsics to tackle the difficult problem of predicting birds-eye-view representation received a large amount of interest recently. A common approach [1], [3], [10], [11], [25], [26] is to use inverse perspective mapping (IPM) to map front-view image onto the ground plane via homography projection. OCT-Net [14] projects a fixed volume of voxels onto multi-view images to collect features and complete 3D detection on the bird’s-eye-view feature representation. LSS [27] is proposed to infer birds-eye-view representation by lifting each image into a frustum of features and collapsing all frustum into a rasterized bird’s-eye-view grid. LIFT [37] projects point clouds and images to BEV planes to compute sparse grid-size attention. BEVFormer [38], [39] exploits both spatial and temporal information through predefined grid-shaped BEV queries. Petr [41], [42] utilizes temporal information of previous frames to boost 3D detection. Voxel Transformer [47] proposes the sparse voxel module and the submanifold voxel module, which can operate on the empty and non-empty voxel positions effectively. Voxel R-CNN optimally balances accuracy and computational efficiency by utilizing voxel representation.

These multi-view detection approaches discussed above primarily address the autonomous driving context in which these sensors are located in proximity to one another with minimal overlapping fields of view. Conversely, in the domain of multi-view 2D/3D detection for pedestrian monitoring, a key research challenge lies in optimally utilizing multi-view information to mitigate occlusion issues in crowded scenes. In the setting, the sensors are positioned at considerable distances from each other and perspective views still exhibit substantial overlapping fields of view.
Fig. 2. System workflow. Given input images of shape \([3, H_i, W_i]\) from \(N\) cameras, we use a convolutional neural network (CNN) backbone to extract multi-scale image features, denoted from low ("l"), medium ("m") to high resolution ("h") as \([N, C_l, H_l, W_l]\), \([N, C_m, H_m, W_m]\), and \([N, C_h, H_h, W_h]\), respectively. To perform voxelized 3D feature aggregation, we create a cuboid made of smaller cubes or voxels (red), where we use a grid size of \(156 \times 156 \times 4\) as an example. We project each voxel to multi-scale image planes, and within each projected voxel, we align and pool the corresponding 2D features along the vertical direction, so that we obtain \(N\) stacks of feature maps of size \([1, C_g, H_g, W_g]\) coming from \(N\) cameras, where subscript "g" means "global". We concatenate and collapse these feature maps along the channel dimension to be used for final 3D prediction. Four output heads are used during prediction. A confidence map and an offset map are predicted to jointly localize objects. The other two heads are responsible for orientation and dimension estimation.

C, H, W represent the number of channels, image height and width, respectively.

III. Method

This section details the proposed system that mainly deals with projection distortions when aggregating multi-view features. There are three components, including multi-scaled feature extraction (Section III-A), voxelized feature aggregation (Section III-B) and multi-branch estimation (Section III-C and Section III-D).

A. Feature Map Extraction

In VFA, first, given \(N\) images of size \([H_i, W_i]\) as input (\(H_i\) and \(W_i\) denote the image height and width respectively and \(N\) is determined by the number of cameras), the proposed network chooses ResNet-18 [30] as feature extractor to generate a hierarchy of multi-scale 2D feature maps from each input image. This CNN calculates multi-scale feature maps separately for \(N\) input images while sharing weights among all calculations. After extracting features, a multi-scale feature representation will be generated, including \([N, C_l, H_l, W_l]\), \([N, C_m, H_m, W_m]\), \([N, C_h, H_h, W_h]\). The subscript \(l\), \(m\) and \(h\) denote low, middle and high-resolution feature representation. \(C_l, H_l\) and \(W_l\) represent the channel, height, and width of low-resolution feature. Before projection, we adjust the number of channels of different scaled features to be consistent through a \(1 \times 1\) convolution, with height and width remaining unchanged.

B. Voxelized Feature Aggregation

The essence of VFA is to map multi-view image features \(F(u, v) \in \mathbb{R}^n\) to corresponding voxel grids, generating 3D voxel feature \(G(x, y, z) \in \mathbb{R}^n\). The voxel grids will be defined beforehand, whose size depends on the proportion of movement range of all objects and grid height is determined by the height of the objects. To explore the impact of size of voxel grids on model performance, extensive experiments are conducted in Section IV-B. For a given voxel grid location \((x, y, z) \in \mathbb{R}^n\), voxel feature \(G(x, y, z)\) is corresponding to the pooling result of the 2D projection of voxel on image feature. The pooling area can be estimated as the bounding box of voxel 2D projection. How VFA aggregates multi-view features can be summarized in the following steps:

1) Before inference, a voxel grid will be generated where each voxel is located at \((x, y, z) \in \mathbb{R}^n\) and with the size
2) Project eight corners of each voxel to multi-view image planes. Given a 3D world position \((x, y, z)\), we can calculate its projected image coordinate \((u, v)\) following,

\[
\gamma \left( \begin{array}{c} u \\ v \\ 1 \end{array} \right) = P_b \left( \begin{array}{c} x \\ y \\ z \\ 1 \end{array} \right) = K[R]t \left( \begin{array}{c} x \\ y \\ z \\ 1 \end{array} \right),
\]

where \(P_b\) denotes projection matrix, \(K, [R]t\) denote intrinsic parameter matrix and rotation-translation matrix respectively, and \(\gamma\) is an arbitrary scale factor. For each voxel, we will get eight projected corners \(c_i(u_i, v_i), i \in \{1, \ldots, 8\}\), on image plane.

3) Calculate the bounding box with the top-left corner \((u_{\text{min}}, v_{\text{min}})\) and bottom-right corner \((u_{\text{max}}, v_{\text{max}})\) from 8 projected voxel corners.

4) Extract the learned feature by average pooling over the bounding box of voxel 2D projection and assign to the corresponding location in the voxel feature \(G\):

\[
G(x, y, z) = \frac{\sum_{u=\min}^{\max} \sum_{v=\min}^{\max} F(u, v)}{(u_{\text{max}} - u_{\text{min}})(v_{\text{max}} - v_{\text{min}})}.
\]

As shown in Fig. 3, multi-view image features are projected to appropriate location in 3D voxel grids. Directly applying 3D Convolutional Neural Networks (3D CNNs) on compact 3D voxel feature representation will cost huge computational resources. Thus, before multi-branch predicting, the 3D voxel feature will be collapsed along the vertical axis, becoming 2D BEV features.

**Difference from and advantage over [11].** Both our method and [11] aim to address the projection distortion problem. To this end, Song et al. [11] generate multiple ground planes at different heights onto which the 2D features are projected. This method, while alleviating overall distortions to some extent, still suffers from distortion on plane. In comparison, through identifying the vertical axis and projecting it onto 2D images, VFA resolves the projection problem in a more thorough way. As to be shown in Fig. 6 A, we observe consistent improvement when implementing the VFA module on top of [11].

### C. Oriented Gaussian Distribution for Ground-truth Modeling

While some methods [6, 9] use pixel-based probability confidence maps to denote object positions (Fig. 4 A), and others [1], [28] employ smooth Gaussian distribution for object location (Fig. 4 C), our method intuitively uses oriented Gaussian distribution for elongated objects. As depicted in Fig. 4 E, the Gaussian distribution’s center, size, and rotation angle corresponds to the object’s BEV plane location, size, and orientation. Given a set of ground truth objects at \([x_i, y_i, z_0]^T\), (all objects are on the ground, so \(z_0 = 0\) by default), with

- \(l_i\) length
- \(w_i\) width
- \(h_i\) height
- \(\alpha_i\) orientation

of \((l, w, h)\).

\[
\Delta_{\text{pos}}(x, y) = \left[ \frac{x}{T} - \frac{l_i}{2} \right] \left[ \frac{y}{T} - \frac{w_i}{2} \right] \left[ \frac{z}{T} - \frac{h_i}{2} \right]^T,
\]

\[
\Delta_{\text{pos}}(x, y) = \left[ \frac{x}{T} - \frac{l_i}{2} \right] \left[ \frac{y}{T} - \frac{w_i}{2} \right] \left[ \frac{z}{T} - \frac{h_i}{2} \right]^T,
\]

\[
\Delta_{\text{pos}}(x, y) = \left[ \frac{x}{T} - \frac{l_i}{2} \right] \left[ \frac{y}{T} - \frac{w_i}{2} \right] \left[ \frac{z}{T} - \frac{h_i}{2} \right]^T,
\]

where \(P_b\) denotes projection matrix, \(K, [R]t\) denote intrinsic parameter matrix and rotation-translation matrix respectively, and \(\gamma\) is an arbitrary scale factor. For each voxel, we will get eight projected corners \(c_i(u_i, v_i), i \in \{1, \ldots, 8\}\), on image plane.

3) Calculate the bounding box with the top-left corner \((u_{\text{min}}, v_{\text{min}})\) and bottom-right corner \((u_{\text{max}}, v_{\text{max}})\) from 8 projected voxel corners.

4) Extract the learned feature by average pooling over the bounding box of voxel 2D projection and assign to the corresponding location in the voxel feature \(G\):

\[
G(x, y, z) = \frac{\sum_{u=\min}^{\max} \sum_{v=\min}^{\max} F(u, v)}{(u_{\text{max}} - u_{\text{min}})(v_{\text{max}} - v_{\text{min}})}.
\]

As shown in Fig. 3, multi-view image features are projected to appropriate location in 3D voxel grids. Directly applying 3D Convolutional Neural Networks (3D CNNs) on compact 3D voxel feature representation will cost huge computational resources. Thus, before multi-branch predicting, the 3D voxel feature will be collapsed along the vertical axis, becoming 2D BEV features.

**Difference from and advantage over [11].** Both our method and [11] aim to address the projection distortion problem. To this end, Song et al. [11] generate multiple ground planes at different heights onto which the 2D features are projected. This method, while alleviating overall distortions to some extent, still suffers from distortion on plane. In comparison, through identifying the vertical axis and projecting it onto 2D images, VFA resolves the projection problem in a more thorough way. As to be shown in Fig. 6 A, we observe consistent improvement when implementing the VFA module on top of [11].

### C. Oriented Gaussian Distribution for Ground-truth Modeling

While some methods [6, 9] use pixel-based probability confidence maps to denote object positions (Fig. 4 A), and others [1], [28] employ smooth Gaussian distribution for object location (Fig. 4 C), our method intuitively uses oriented Gaussian distribution for elongated objects. As depicted in Fig. 4 E, the Gaussian distribution’s center, size, and rotation angle corresponds to the object’s BEV plane location, size, and orientation. Given a set of ground truth objects at \([x_i, y_i, z_0]^T\), (all objects are on the ground, so \(z_0 = 0\) by default), with
We use scaling factor $\gamma$ to resize the original movement range to a confidence map. For example, in the MultiviewC dataset, $\gamma = 25$. We apply focal loss [31] to the confidence map $S(x, y)$ to mitigate class imbalance and regress object occupancy. Two additional networks are used for orientation and dimension prediction. Rather than using multi-bins for angle range and offset, we use circle smooth label (CSL) [32] for angle prediction.

$$CSL(x) = \begin{cases} g(x), & \theta - r < x < \theta + r \\ 0, & \text{otherwise} \end{cases},$$

In the equation, $g(x)$ is a Gaussian window function with radius $r = 6$, and $\theta$ is the predicted angle. Leveraging oriented Gaussian distribution’s orientation retention, the orientation head predicts grid cell orientations on the bird’s-eye-view plane using CSL.

For dimension head, we predict the scale offset $\Delta_{\text{dim}}(x, y)$, which is given by,

$$\Delta_{\text{dim}}(x, y) = \begin{bmatrix} \log \frac{d_1}{\bar{d}} & \log \frac{w_1}{\bar{w}} & \log \frac{h_1}{\bar{h}} \end{bmatrix}^T,$$

where $d_i = [l_i, w_i, h_i]$ denotes the dimension of targets and $\bar{d} = [\bar{l}, \bar{w}, \bar{h}]$ represent the mean dimension over all objects.

IV. EXPERIMENT
A. Experimental Setting
In this section, we evaluate the performance of the proposed VFA on WildTrack [2], MultiviewX [1], MVM3D [3] and MultiviewC datasets.

1) Datasets: Existing multiview detection datasets. The WildTrack dataset is a real-world dataset while the MultiviewX dataset is a synthetic dataset for multiview pedestrian detection. Both datasets do not include the orientation label of the object. Therefore, the validation on these datasets is focus on occlusion cases. The MVM3D dataset for mobile robot detection is similar to our proposed dataset, containing images collected from multiple views, bounding boxes including position and orientation, and obstacles with different heights. However, it does not consider targets with diverse sizes.

Proposed MultiviewC dataset diversifies multiview 3D detection, created using Unreal Engine 4.26. Set in a 39×39 meter cattle farm under sunny conditions, it includes 3,920 images (720×1280 pixels each) from 7 strategically placed calibrated cameras. It focuses on 25 cows performing five actions—sleeping, grazing, running, walking, and laziness—simulating occlusions by pausing movements upon encounters. This unique dataset covers varied target sizes, aspect ratios, and actions affecting size. Cow dimensions range from 1.1 – 1.5 meters in height and width, and 2.48 – 2.78 meters in length, with action-induced dimension changes shown in Fig. 5.

MultiviewC engine data collection features include real-time video recording and customizable data capture, with adjustable acquisition frequency. It records detailed information on the rotation, dimension, position, and actions of target cattle, storing this data in txt files. The engine supports simultaneous multi-view capture for 50 players online via LAN, with synchronized operations and local data storage. Flexible perspective switching among eight fixed camera locations (excluding camera 8) ensures comprehensive data collection from multiple viewpoints. Its real-time capabilities enable precise data annotation, making it ideal for analyzing cattle movements and behaviors.

2) Evaluation Metrics: Following the existing methods such as [1], [3], [6], we adopt Multi-Object Detection Precision (MODP) and Multi-Object Detection Accuracy (MODA), recall, precision as the evaluation metrics for the multiview 2D detection, whereas MODA accounts for both the false positive and false negatives and MODP focus on the precision of detected object. We use MODA and MODP as primary performance indicator in multi-view 2D detection. For multi-view 3D detection, Average Precision of 3D detection (AP3D) [7], Average Orientation Similarity (AOS) and Orientation Score (OS) [8] are selected. These metrics evaluate orientation and dimension, while multi-view 2D detection only evaluates position.

3) Implementation Details: During experiments, input images of other sizes are resized to 720×1,280 pixel resolution to save memory. We employ a multi-scale feature extractor based on a ResNet-18 taking the output of the last three layers as the multi-scale feature. After feature extraction, before
B. Evaluation of VFA

Comparison with the state of the art. Results on Wildtrack dataset and MultiviewX dataset: The evaluation is focused on how our proposed method deals with occlusion cases. A comparison between our VFA method and previous methods on multiview 2D detection is shown in Table I. On the Wildtrack dataset, VFA achieves competitive results compared to the state-of-the-art SHOT [11], notably exhibiting an +18.1% increase on MODP. On the MultiviewX dataset, we achieved the highest performance, also showing a significant improvement on MODP metric. It is worth mentioning that the proposed voxel-based method performs well in the precision of prediction. This also shows that their projection-based method [1], [3], [11] is good at suppressing false positives, but sometimes has a tendency to miss a few targets. On MVM3D dataset and MultiviewC dataset, we focus on 3D detection, including position, dimension and orientation. The comparison between our proposed VFA method with the MVM3Det method and a single view method, Deep3DBox [8] is shown in Table II. Our VFA method is trained with 3 different encoding distributions: point distribution (PD), Gaussian distribution (GD), and oriented Gaussian distribution (OGD). The results show that although on par on the MVM3D dataset, our VFA method outperforms Deep3DBox and MVM3Det by large margins on all metrics on the MultiviewC dataset. The reason is that the MultiviewC dataset contains cow elongated bodies and different poses such as standing and lying, therefore more challenging than the MVM3D dataset. In the case of encoding methods, our proposed OGD encoding can further enhance performance.

Comparison with [11]. The SHOT method [11] is the closest work to ours, which also addresses the projection distortion problem. To demonstrate the effectiveness of our proposed VFA, we change SHOT by replacing its 2D-3D projection with VFA component and call this as “SHOT with VFA”. We then compare SHOT and “SHOT with VFA” under the same experiment settings on the MultiviewX dataset. Fig. 6 A shows that the “SHOT with VFA” method outper-
forms the original SHOT on all metrics.

Hyper-parameter analysis. There are two primary hyperparameters of VFA which are grid height and number of voxel layers. First, we assess the influence of grid height as shown in Fig. 6 B (and Table 4 in the Supplementary Material). It is clear that our VFA performs best with the height of 1.6 meters and the grids size of $0.1 \times 0.1 \times 0.2$ for both Wildtrack and MultiviewX datasets. We then find the optimal number of voxel layers while the grid height is fixed to 1.6 meters as shown in the Fig. 6 C (and in Table 4 of Supplementary Material). Results show that more layers achieve better performance as well as higher computational complexity, and 8 layers seems to be a good balance.

C. Further Analysis

Influence of different number of input views. In order to evaluate the impact of view number, we tested 1, 2, 4, and 7 camera views, using only the corner cameras in the first three cases. Table III shows that our method significantly improves from single to multiple views, confirming VFA's ability to aggregate multiview features and mitigate occlusions. However, the performance difference between two and four views is marginal, as the additional corner cameras provide limited useful features for detection. Utilizing all camera views further improves performance, as the three cameras above the trough offer additional multiview cues.

Effectiveness of oriented Gaussian distribution. In order to evaluate the validity of our proposed RGD labeling method, we record the training process of RGD labeling and GD labeling. Table II shows that VFA using oriented Gaussian distribution achieves better precision than using point and Gaussian distributions. We notice that all initial predictions tend to share a similar pattern that strong blobs are oriented and naturally aligned with the direction of the targets. Therefore, it is sensible that the orientation should be encoded in the aggregated features and predictions. Thus, instead of point distribution or Gaussian distribution, we train our VFA network using labels encoded by oriented Gaussian distribution. Finally, Fig. 7 demonstrates that while VFA achieves comparable losses with all three encoding distributions after sufficient training, the oriented Gaussian distribution exhibits remarkably faster convergence—at least $4 \times$ faster than the alternatives.

V. Conclusions

This paper proposes the voxelized 3D feature aggregation (VFA) method to resolve the problem of projection distortion caused by the usage of 2D transformation schemes. Our method, in a straightforward way, maps Z-axis-aware 3D voxels onto the 2D multi-view images, allowing features along the same vertical line to be projected onto the same position on the ground plane. VFA purifies the aggregated features on the ground plane and thus improves the system accuracy. On a side contribution, we use an oriented Gaussian modeling method for objects that do not have a square ground-truth on the ground plane, like cattle. This method allows for more efficient training process. We validate this system on four datasets (including a newly introduced cattle detection dataset), where we report very competitive detection accuracy compared with the state-of-the-art system.
