Tracking Livestock Using a Fully Connected Network and Kalman Filter

Farah Sarwar\textsuperscript{1}, Anthony Griffin\textsuperscript{1(✉)}, and Timotius Pasang\textsuperscript{2}

\textsuperscript{1} Electrical and Electronic Engineering Department, Auckland University of Technology, Auckland 1142, New Zealand
\{farah.sarwar,anthony.griffin\}@aut.ac.nz
\textsuperscript{2} Mechanical Engineering Department, Auckland University of Technology, Auckland 1142, New Zealand
timotius.pasang@aut.ac.nz

Abstract. Multiple object tracking (MOT) consists of following the trajectories of different objects in a video with either fixed or moving background. In recent years, the use of deep learning for MOT in the videos recorded by unmanned aerial vehicles (UAVs) has introduced more challenges and hence has a lot of room for extensive research. For the tracking-by-detection method, the three main components, object detector, tracker and data associator, play an equally important role and each part should be tuned to the highest efficiency to increase the overall performance. In this paper, the parameter selection of the Kalman filter and Hungarian algorithm for sheep tracking in paddock videos is discussed. An experimental comparison is presented to show that if the detector is already providing good results, a small change in the system can degrade or improve the tracking capabilities of remaining components. The encouraging results provide an important step in an automated UAV-based sheep tracking system.

Keywords: Livestock · UAV · Tracking · CNN · Kalman filter

1 Introduction

Research in the field of object tracking—especially multiple object tracking (MOT)—is gaining more and more attention with the increase in demand to automate the systems for crowd estimation, facial recognition, wildlife monitoring, autonomous driving, vehicle detection and surveillance, to name a few. Some applications, like tracking eye movement of drivers for fatigue detection, require the detection and tracking of a single object, while many others like autonomous driving, pedestrians detection, and livestock tracking have to simultaneously track multiple objects in the consecutive video frames. Irrespective of single object tracking, an important step in MOT is to associate the track identities (IDs) with detection, known as data association. This step is needed to keep a track of all those objects that enter or leave any frames and also to keep the IDs associated with respective objects only, especially in occlusion [8].

© Springer Nature Switzerland AG 2021
M. Nguyen et al. (Eds.): ISGV 2021, CCIS 1386, pp. 247–261, 2021.
https://doi.org/10.1007/978-3-030-72073-5_19
MOT can be performed as an online or offline task; online tracking [6] is usually done for real-time applications where future frames are not available and information from the previous frames is used to predict objects’ motion, while offline tracking [15] is usually performed for recorded videos that can utilise the information from past and future frames. However, in both cases, the main challenges are the presence of noise in a video recorded from a low-resolution camera, complex object contour, object occlusion, variation in object scale and blurriness. The use of an unmanned aerial vehicle (UAV) adds more challenges, like a sudden change in the movement pattern of an object due to the fluctuations in the UAV flight in a windy environment and increase in scene complexity due to moving background along with objects. The object size also varies with changes in UAV altitude and results in the loss of most of the distinguishing features. As images and videos from different perspectives are needed for object detection and tracking—rather than using fixed position cameras—the use of UAVs is gaining similar interest in various research areas of computer vision and artificial intelligence.

In this research article, we propose a tracking-by-detection method for livestock detection, tracking and counting using a UAV. The data was collected at different heights using DJI Phantom 3 Pro in sheep farms near Pirinoa, New Zealand. In this article, the results of videos recorded from a height of 80 m are reported. The task of sheep detection was performed using a U-Net model [13,14]. For livestock tracking, a Kalman filter [17] as a motion predictor and Hungarian algorithm [9] as a detection-to-track linker were used. It was observed that as all sheep look tiny and similar from this height, a careful parameter selection is needed for the Kalman filter and Hungarian algorithm to predict the next state close to the actual location. Although the object detector has the main impact on such systems, the adjustment of the tracker and data associator parameters is equally important and a small change can increase the overall efficiency.

2 Related Work

Tracking algorithms are broadly classified in two main categories; (i) detection-based tracking [4] and (ii) detection-free tracking [10]. The detection-based tracking—also known as tracking-by-detection—has a pre-trained object detector to locate objects in each frame as a preliminary step. These values—either as a bounding box or centroid—are used by the tracker to initialize the tracking process. If an object detector fails in its task at any intermediate frame, due to occlusion or any reason, the tracker keeps predicting the object’s state for a few more frames. It is the task of data associator to link the same track with a respective object once it reappears in the video. However, if the object stays undetected for many frames then the track is deleted and a new track is assigned to it later, if needed. Detection-free tracking, on the other hand, requires initialization of the object locations in the first frame and tracks them throughout the video using the objects’ features. It works best for videos recorded by fixed position cameras, which have distinctive objects in the foreground with a non-moving
background. It is not a very useful approach for those videos where objects can enter or leave in the middle of the video.

Siqi Ren, Yue Zhou and Liming He [12] performed MOT by initially dividing detection results into false, high uncertain, and low uncertain categories, and delayed the results of low uncertain detections until the end of the video. They penalized the false detections and constructed a tracking tree for less uncertain detections. This helped them to improve the overall performance of the system. Tubelets with convolutional neural networks (T-CNN) was proposed by Kai Kang et al. [7] as an end-to-end deep learning framework for detection and tracking by incorporating temporal and contextual information. Gaoang Wang et al. [15] also combined temporal and appearance information to generate tracklets by associating detection results in consecutive frames, known as TrackletNet Tracker (TNT). Similarly, Zhongdao Wang et al. [16] proposed a shared MOT model that combined a target detection and appearance model into a single-shot detector in a way that it can simultaneously output detection and update location. Shivani et al. [8] used the Deep Simple Online Real Tracking (Deep SORT) algorithm as a baseline to build their algorithm and used a combination of YOLOv3 and RetinaNet for generating detection results in a video recorded by a drone-mounted camera. Kwangjin Yoon et al. [18] proposed a data association method for MOT using deep neural networks and highlighted the importance of the data association method in tracking-by-detection methods. They used bounding boxes and a track-history as input for long short term memory (LSTM) networks. The final output is an association matrix to show correspondence between tracks and detection. Similarly, a multiple hypothesis tracker (MHT) [3,21] and particle filter [4] can be used with good efficiency but with a comparatively high computational cost. Recent survey papers [5,20] gave a systematic literature review of various MOT algorithms in their four main stages: feature extraction, detection, motion prediction and data association. They highlighted the issues related to MOT and how the complex background and occlusion problems can decrease the performance of any algorithm.

MOT using Kalman filter and different data association methods have been used by many researchers [6,10,19]. To the authors’ knowledge, the literature has revealed that the adjustment of its parameters taking into account the physical context of the objects have not been discussed in detail. Kalman filters have been previously used for tracking large objects in different scenarios. This research article focuses on the parameter adjustment to observe the impact on its performance for tracking many small and similar objects, in videos recorded by a UAV-mounted camera. Instead of using the bounding boxes, the centroid values of the respective sheep are used.

3 Methodology

In the tracking-by-detection method, the object detector plays a vital role in keeping up the overall efficiency of the system. By improving the performance of the detector, it becomes easier for the tracker and data associator to estimate
Fig. 1. A system diagram of livestock detection, tracking and data association, starting from the second frame of the video. A Kalman filter provides the predicted states of the tracks and Hungarian algorithm use these predicted state values and object detector’s estimations to link tracks with detected objects. Correction of tracks is the last step of this system and all tracks are predicted for the upcoming frame in the next time step.

motion and associate detections with existing tracks, respectively. The detector used in this research is a U-Net model to detect sheep in videos recorded by a UAV, which can detect sheep with a very high recall [14] and gives the centroid values of all the detected objects. Offline tracking is performed and Fig. 1 shows the flowchart of our object tracking methodology. It is further explained in the following subsections.

3.1 Object Tracker

Some pre-processing steps were performed to reduce the overall tracking time. It was assumed that the fenced area was pre-defined in video frames and no objects were detected outside the paddock fence. The recorded videos—as well as video frames—were downsampled by a factor of two, and every second frame was fed to the tracker and object detector. The original dimension of each RGB frame was 4096 × 2160 pixels and was reduced to 2048 × 1080 pixels. After these pre-processing steps, the first frame was only used by the object detector to estimate the centroid of each object in the first frame and initialize the Kalman filter accordingly. Figure 1 shows the process from the second frame onward.

The trained U-Net model [14] was used to detect sheep in each frame of the video, and provides two outputs: a probability map and an object count. The mean shift clustering algorithm is applied to the probability map to compute the centroids of the detected sheep in the respective frame. These centroid values, and the object’s velocity in x and y directions, were provided to the Kalman filter [17] as the initial value of the object’s state. Although, the coordinates
are directly provided by the detector, the velocity can be provided in two ways. Either by setting the initial velocity to zero and using higher covariance values for the filter, or using the average velocity of livestock obtained from the information in previous frames. The second option is only valid where all objects are moving with the same velocity, and the same value can be used for all objects. To observe how the tracker adjusts the states in a noisy environment, the results are presented by setting initial velocity to zero. So, for the \( m \)-th object in the \( i \)-th frame, the state vector is provided as

\[
s^m_i = [c^m_{x,i}, v^m_{x,i}, c^m_{y,i}, v^m_{y,i}]^T,
\]

(1)

where \( c^m_{x,i} \) and \( c^m_{y,i} \) are the \( x \)- and \( y \)-coordinates of the true centroid of the \( m \)-th object, and \( v^m_{x,i}, v^m_{y,i} \) are its velocities in the respective directions. For each object, the state in the \( i \)-th frame can be modelled using the value of the state of the respective objects in the \((i-1)\)-th frame as follows

\[
s_i = Fs_{i-1} + w_i,
\]

(2)

where \( F \) is the state transition matrix and \( w_i \) is the process noise, which is zero mean normal distribution with covariance \( Q \). The state transition model is the core of the filter and needs to be designed carefully according to the physical context and motion properties of the tracked object. This is a very crucial step and is a main factor in predicting the next state from the current state. \( Q \) accounts for the unexpected noise in the whole system. For example, in the case of a UAV, a smooth flight can experience disturbances due to gusty wind or the UAV speed may need adjustments repeatedly, hence causing unexpected variations in object motion. This value can be adjusted according to the quality of recorded video and how much variation appeared in the object’s motion.

For each frame, a measurement related to each state can be modelled as

\[
z_i = Hs_i + v_i,
\]

(3)

where \( H \) is the measurement model matrix and \( v_i \) is the observation noise and is also assumed to be zero mean normal distribution with covariance \( R \). Like \( Q \), \( R \) should be adjusted as per the uncertainty of the detector’s output and small values should be used if the detector’s estimations are very accurate. The Kalman filter is a recursive algorithm, that predicts the optimal states before reading the observation, and updates the states using the values provided by the detector for the current frame.

### 3.2 Data Association

The assignment problem was solved using the Hungarian algorithm [9] to link the predicted states of the tracker with the detector’s estimations for the current
frame. It uses a cost matrix that is computed between each value provided by
the detector and the existing tracks. A cost of non-association (CNA) needs
to be provided to adjust the leniency of this assignment task and was adjusted
experimentally. This helped to assign the existing track IDs to the same detected
object, and identify unassigned detections and tracks. The assigned tracks were
sent to the correction step of the filter to correct and update trackers’ states.
New tracks were created for the unassigned detections, while unassigned tracks
were checked against a few conditions to decide whether they should be deleted
or kept in the loop.

Fig. 2. A visual overview of how a video of paddock is recorded using a UAV, with
white ellipses representing sheep. As the UAV moves forward, sheep exit at the bottom
of the current frame, and new sheep enter at the top.

3.3 New and Old Tracks

As the UAV flies over the paddock, many objects leave the frame from the
bottom and many enter the frame through the top, as illustrated in Fig. 2. As
this happens, new tracks need to be created with unique IDs and a few need to
be deleted accordingly. Tracks were terminated using one of the following three
conditions:

1. The visibility of the track was less than 60% and it was visible for less than
two frames,
2. The predicted state of the track was outside the frame dimension,
3. The object linked with the track was not detected consecutively in more than $F_{\text{lost}}$ frames.

The first condition was used to terminate false positive (FP) detections as they appear only a few times in the video, and the second condition was for all those objects which left the frame from the bottom of the video. Such tracks should be deleted as early as possible to avoid an unbounded growth of tracks and reduce computational resources. However, there were some cases where a sheep, standing in a group, was detected in a frame and then the detector was unable to locate it in some consecutive frames. In such situations, the track was kept in the loop for at least $F_{\text{lost}}$ frames and was then deleted afterwards. The third condition was for these kinds of objects and was very crucial.

4 Experimental Evaluation

4.1 Parameter Selection

The height of 80 m for the UAV flight was chosen to keep the paddock fences within a frame from the left and right sides. The UAV moved in one direction only so the objects—sheep—exhibit linear motion of constant velocity. A few sheep showed random non-linear motion in some frames, and the parameters of the Kalman filter and Hungarian algorithm were tuned to cover such issues. The values of the variables used in the prediction step were as follows:

$$
F = \begin{bmatrix}
1 & T_x & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & T_y \\
0 & 0 & 0 & 1
\end{bmatrix}, \quad Q = q I_{4\times 4}, \quad R = r I_{4\times 4}, \quad (4)
$$

where $I_{4\times 4}$ is the $4 \times 4$ identity matrix, and $T_s$ is the sampling interval which was set to 1, and is used to keep the velocity of the object in the respective direction constant. The values of $q$ and $r$ were varied from $[1, 2, 3, 4, 5, 10]$, to see their impact on object tracking. Usually larger values of these covariances ensure that each track is always linked with some detection but this can worsen the condition of ID switching and, hence can degrade the tracking performance. As one of the goals was to count all the sheep in a paddock, a trade-off point needed to be selected between the creation of multiple tracks for a missed object within frame boundaries, or keeping the track until the object appeared again. If a track is deleted and created for such objects this increases the number of false negatives (FNs) in the system. However, the opposite case increases the number of false positives (FP), especially if the predicted location by the tracker is not close to the actual object. To observe this, $F_{\text{lost}}$ was tested for values of $[2, 4, 6, 8, 10]$ frames and the impact of variation among these values is discussed shortly.
4.2 Metrics

There are many metrics defined to evaluate the performance of multiple object tracking [1,2,11], and each one of them covers different aspects of the overall efficiency. The standard MOT evaluation metrics were used to report results in this article. However, it was difficult to compare results with any other researchers’ work, as there are no publicly available datasets of livestock that can be used by others working in our field.

MOT precision (MOTP), MOT accuracy (MOTA) and FP rate (FPR) were used as the main metrics. MOTP measures the tracker’s precision as a motion predictor and is defined as

\[
\text{MOTP} = 1 - \frac{\sum_i d(\hat{c}_i, c_i)}{\sum_i M(i)},
\]

where \( \hat{c}_i \) and \( c_i \) are \( 2 \times 1 \) vectors containing the x- and y-coordinates of the predicted and ground truth centroids of the objects, respectively. \( d(a, b) \) is the Euclidean distance between \( a \) and \( b \), and \( M(i) \) is the number of the matches between \( \hat{c}_i \) and \( c_i \). The closer the predicted and true centroids of the sheep, the lower the Euclidean distance between them, and hence the higher the MOTP.

The MOTA gives a measure of all the errors that occurred during the whole tracking process, and is computed as

\[
\text{MOTA} = 1 - \frac{\sum_i [\text{FP}(i) + \text{FN}(i)]}{\sum_i \text{GT}(i)},
\]

where \( \text{FP}(i) \), \( \text{FN}(i) \) and \( \text{GT}(i) \) represent the number of false positives, false negatives, and ground truth values in the \( i \)-th frame, respectively. The FPR measures the number of FPs as a fraction of the total objects in all the frames, defined as

\[
\text{FPR} = \frac{\sum_i \text{FP}(i)}{\sum_i \text{GT}(i)}.
\]

To count the total livestock in a paddock, the sheep that leave the frame from lower boundary are counted and then the sheep count from the last frame are added.

4.3 Results

Videos were recorded from the same paddock in sunny and cloudy weather, and Fig. 3 shows the first frame from each video. The respective paddock had a total of 352 sheep in it and the recorded videos had between 100 to 250 sheep in each frame. Although the actual sheep count was the same in both videos, the livestock spread was different—as the recording was done on different days—and so there is a difference in the total ground truth values for the respective videos. Thus the cumulative ground truth was 15045 and 19798 in the overcast and sunny videos, respectively.
**Fig. 3.** First frames of videos recorded by the UAV at 80 m altitude in (a) sunny and (b) cloudy weather.

**Table 1.** Tracking performance metrics for various values of the process and measurement noise covariances, $q$ and $r$. CNA and $F_{\text{lost}}$ were set at 20 and 10, respectively. For both weather conditions the total sheep in the paddock were 352.

| (a) Cloudy Conditions – Ground truth matches are 15045 | Metrics | Sheep count | Total FP | Total FN | Total M | FPR | MOTP | MOTA |
|---|---|---|---|---|---|---|---|---|
| Noise covariances | | | | | | | | |
| $q$ and $r$ | | 14989 | 176.61% | 97.61% | −76.98% |
| 1 | 865 | 26571 | 56 |
| 2 | 786 | 21564 | 54 |
| 3 | 753 | 16492 | 52 |
| 4 | 704 | 14055 | 49 |
| 5 | 690 | 12065 | 49 |
| 10 | 528 | 5224 | 50 |

| (b) Sunny Conditions – Ground truth matches are 19798 | Metrics | Sheep count | Total FP | Total FN | Total M | FPR | MOTP | MOTA |
|---|---|---|---|---|---|---|---|---|
| Noise Covariances | | | | | | | | |
| $q$ and $r$ | | 19187 | 3.33% | 92.92% | 93.58% |
| 1 | 384 | 660 | 611 |
| 2 | 353 | 322 | 599 |
| 3 | 352 | 249 | 601 |
| 4 | 352 | 250 | 602 |
| 5 | 351 | 252 | 608 |
| 10 | 350 | 235 | 616 |

The effect of selecting values for parameters such as process and observation noise covariance, CNA and $F_{\text{lost}}$ was observed experimentally. Here, we will refer to $q$ and $r$ both as noise covariance because both values were changed simultaneously. In the first case, values of $F_{\text{lost}}$ and CNA were kept constant at 10 and 20, respectively, and the covariance was varied through $[1, 2, 3, 4, 5, 10]$. Values of $F_{\text{lost}}$ and CNA were selected to keep the system a bit relaxed in terms of deleting the invalid tracks, and data assignment task respectively. The variation in the performance of system during this process is shown in Table 1. For the video recorded in cloudy weather the value of MOTA was negative for lower values of the covariance. New track IDs were created repeatedly for the existing tracks, and this kept on increasing the cumulative values of FP and FN with each new frame and resulted in a higher value of the numerator in Eq. 6. Hence,
MOTA was negative for lower values of noise covariance, but this was not the case for the other video. One of the reasons behind higher rate of new tracks creation for the first video is that the Kalman filter was unable to predict the next location of the sheep properly and the lower CNA was letting the filter create new tracks instead of linking them with existing ones. We were observing multiple ID switches in both videos for higher values of covariance, and a value of 2 was used while observing the effect of variation in other parameters on the system.

Table 2. Tracking performance metrics for various values of CNA. Values of \( q \), \( r \) and \( F_{\text{lost}} \) were set at 2, 2 and 10, respectively. For both weather conditions the total sheep in the paddock were 352.

(a) Cloudy Conditions – Ground truth matches are 15045

| CNA | Metrics | Sheep count | Total FP | Total FN | Total M | FPR | MOTP | MOTA |
|-----|---------|-------------|----------|----------|---------|------|------|------|
| 20  | 786     | 21564       | 54       | 14991    | 143.33% | 97.66% | -43.69% |
| 40  | 361     | 105         | 47       | 14998    | 0.70%   | 97.60% | 98.99% |
| 60  | 353     | 46          | 43       | 15002    | 0.31%   | 97.48% | 99.41% |
| 80  | 353     | 37          | 42       | 15003    | 0.25%   | 97.41% | 99.47% |
| 100 | 353     | 37          | 42       | 15003    | 0.25%   | 97.41% | 99.47% |

(b) Sunny Conditions – Ground truth matches are 19798

| CNA | Metrics | Sheep count | Total FP | Total FN | Total M | FPR | MOTP | MOTA |
|-----|---------|-------------|----------|----------|---------|------|------|------|
| 20  | 353     | 322         | 599      | 19199    | 1.63%   | 92.70% | 95.35% |
| 40  | 350     | 191         | 595      | 19203    | 0.96%   | 92.74% | 96.03% |
| 60  | 350     | 191         | 595      | 19203    | 0.96%   | 92.74% | 96.03% |
| 80  | 350     | 190         | 595      | 19203    | 0.96%   | 92.74% | 96.03% |
| 100 | 350     | 193         | 601      | 19197    | 0.97%   | 92.80% | 95.99% |

Next the values of noise covariance and \( F_{\text{lost}} \) were fixed at 2 and 10 respectively, while CNA was varied between \([20, 40, 60, 80, 100]\). The higher values of CNA improved the system’s efficiency, shown in Table 2, as new tracks were only created for those sheep that enter the frame from the top or those which were not detected by the object locator in the previous few frames. For both videos, little variation was observed above the value of 60 for CNA, and as per the presented results, the value of 80 was considered to be a good option for both cases. However, as tracks for the undetected objects were still allowed to stay for at least 10 frames, a reduction in this value can reduce the number of FPs and FNs.

The next observations were made using the noise covariance value of 2, CNA equal to 80 and \( F_{\text{lost}} \) varied on \([2, 4, 6, 8, 10]\). The results with these settings for
both videos are shown in Table 3. The system’s response is slightly different for both videos and the MOTA decreased if tracks were kept for more than 4 and 2 frames in the video recorded in cloudy and sunny weather, respectively. The best values of MOTA were observed to be 99.55% and 96.26%. For the second video, as the detector was not detecting a few sheep in some consecutive frames, it caused higher a cumulative FN for all the cases. Still, 96.26% shows a good performance of the overall system. The experimental evaluation shows that the best selection of the covariance values in the Kalman filter for tracking small objects should be close to the radius of tracked objects. Higher values may lead to an ID switch or mismatch error. Also, even if the Kalman filter is at its best performance, the data linker can degrade the performance by not linking the tracks with corresponding detected objects when the value of CNA is less. The main parameter of our concern is MOTA as it covers the different types of errors made by the tracking system, and it was improved by reducing the $F_{\text{lost}}$ value, as shown in Table 3.

Table 3. Tracking performance metrics for various values of $F_{\text{lost}}$. The process and measurement noise covariances, $q$ and $r$, and CNA were set at 2, 2 and 80, respectively. For both weather conditions the total sheep in the paddock were 352.

| Threshold $F_{\text{lost}}$ (frames) | Metrics          | Sheep count | Total FP | Total FN | Total M | FPR  | MOTP  | MOTA  |
|-------------------------------------|------------------|-------------|----------|----------|--------|------|-------|-------|
| (a) Cloudy Conditions – Ground truth matches are 15045 |                  |             |          |          |        |      |       |       |
| 2                                   |                  | 350         | 19       | 53       | 14992  | 0.12%| 97.76%| 99.52%|
| 4                                   |                  | 352         | 25       | 43       | 15002  | 0.17%| 97.53%| 99.55%|
| 6                                   |                  | 352         | 31       | 43       | 15002  | 0.21%| 97.52%| 99.51%|
| 8                                   |                  | 353         | 35       | 42       | 15003  | 0.23%| 97.45%| 99.49%|
| 10                                  |                  | 353         | 37       | 42       | 15003  | 0.25%| 97.41%| 99.47%|
| (b) Sunny Conditions – Ground truth matches are 19798 |                  |             |          |          |        |      |       |       |
| 2                                   |                  | 344         | 75       | 666      | 19132  | 0.38%| 93.77%| 96.26%|
| 4                                   |                  | 346         | 116      | 628      | 19170  | 0.59%| 93.42%| 96.24%|
| 6                                   |                  | 346         | 145      | 611      | 19187  | 0.73%| 93.08%| 96.18%|
| 8                                   |                  | 348         | 172      | 601      | 19197  | 0.87%| 92.77%| 96.09%|
| 10                                  |                  | 350         | 190      | 595      | 19203  | 0.96%| 92.74%| 96.03%|

Figure 4 shows a case of FN and FP from one of the intermediate frames. The blue color shows an FN instance, and the Kalman filter has predicted the location of this sheep at a different location which is shown in cyan color. In this case the system gives one FN and one FP instance. The red color shows that the detector was unable to detect the object, however, the Kalman filter predicted the location correctly in this frame. It is neither an FP nor an FN case and still counted as a matched track. Figure 5 shows that if the sheep is not detected for any less than $F_{\text{lost}}$ frames, the same track ID is assigned to it later. These are the sub-images cropped from four consecutive frames and show two such cases
Fig. 4. The cyan color shows an FP detection, the blue shows an FN instance and red shows that the detector does not estimate this object but tracker estimated the object location perfectly. (Color figure online)

Fig. 5. Cropped sub-images from four successive frames (left to right). In the second frame the sheep with track ID 117 was not detected by the object detector but was tracked successfully by the Kalman filter. In the third frame it was detected again and same track ID was assigned to it. The same issue occurred for the sheep with track ID 116 in the third and fourth frames.

of sheep with IDs 116 and 117. Both of these figures were captured when the values of $q$, $r$, $F_{\text{lost}}$, and CNA were set to 2, 2, 4 and 80 respectively.

Finally, Figs. 7 and 6 show the overlaying graphs of ground truth and matched object counts, which are referred as true count and matched count, respectively. The difference of these values was higher in the case of sunny video as the detector was missing a few sheep in each frame. However, a maximum per frame difference of one and ten was observed in the cloudy and sunny videos, respectively.
Fig. 6. True and matched counts in respective frames for the video recorded in sunny weather, the lower plot highlights the per frame difference between these two values. In this video the object detector failed to detect a few sheep in multiple frames which degraded the performance of the overall system too. A maximum difference of 10 sheep was recorded in 39-th frame of this video.

Fig. 7. True and matched counts in respective frames for the video recorded in cloudy weather, the lower plot highlights the per frame difference between these two values. The object detector used performs better in cloudy weather, and this is illustrated in the maximum tracking error of one being recorded.

5 Conclusion

In this paper, we presented an offline, tracking-by-detection method for livestock tracking in videos recorded by a UAV-mounted camera. We used our previous object detector whose high accuracy made tuning the tracking parameters an achievable task. Parameters of the Kalman filter and Hungarian algorithm were tuned to improve the performance of the overall system. With the best combinations of these parameters, MOTAs of 99% and 96% were achieved in different
weather conditions. However, such performance may not be possible if the object
detector failure rate increases beyond a certain limit.

Future work will include capturing and labelling more videos including ones
at higher altitudes.

References

1. Bernardin, K., Elbs, A., Stiefelhagen, R.: Multiple object tracking performance
metrics and evaluation in a smart room environment. In: 6th IEEE International
Workshop on Visual Surveillance, in Conjunction with ECCV, vol. 90, pp. 91–99
(2006)
2. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance:
the CLEAR MOT metrics. EURASIP J. Image Video Process. 2008, 1–10 (2008).
https://doi.org/10.1155/2008/246309
3. Blackman, S.S.: Multiple hypothesis tracking for multiple target tracking. IEEE
Aerosp. Electron. Syst. Mag. 19(1), 5–18 (2004). https://doi.org/10.1109/MAES.
2004.1263228
4. Breitenstein, M.D., Reichlin, F., Leibe, B., Koller-Meier, E., Van Gool, L.: Robust
tracking-by-detection using a detector confidence particle filter. In: IEEE 12th
International Conference on Computer Vision, pp. 1515–1522. IEEE (2009)
5. Ciaparrone, G., Sánchez, F.L., Tabik, S., Troiano, L., Tagliaferri, R., Herrera, F.:
Deep learning in video multi-object tracking: a survey. Neurocomputing 381
61–88 (2020). https://doi.org/10.1016/j.neucom.2019.11.023
6. Iraei, I., Faez, K.: Object tracking with occlusion handling using mean shift,
Kalman filter and edge histogram. In: 2nd IEEE International Conference on Pattern
Recognition and Image Analysis (IPRIA), pp. 1–6. IEEE (2015). https://doi.
org/10.1109/PRIA.2015.7161637
7. Kang, K., et al.: T-CNN: tubelets with convolutional neural networks for object
detection from videos. IEEE Trans. Circ. Syst. Video Technol. 28(10), 2896–2907
(2017)
8. Kapania, S., Saini, D., Goyal, S., Thakur, N., Jain, R., Nagrath, P.: Multi object
tracking with UAVs using Deep SORT and YOLOv3 RetinaNet detection framework.
In: Proceedings of the 1st ACM Workshop on Autonomous and Intelligent
Mobile Systems, pp. 1–6 (2020). https://doi.org/10.1145/3377283.3377284
9. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Res.
Logistics Q. 2(1–2), 83–97 (1955). https://doi.org/10.1002/nav.3800020109
10. Li, X., Wang, K., Wang, W., Li, Y.: A multiple object tracking method using
Kalman filter. In: The 2010 IEEE International Conference on Information and
Automation, pp. 1862–1866. IEEE (2010). https://doi.org/10.1109/ICINFA.2010.
5512258
11. Milan, A., Schindler, K., Roth, S.: Challenges of ground truth evaluation of multi-
target tracking. In: IEEE Conference on Computer Vision and Pattern Recognition
Workshops, pp. 735–742 (2013)
12. Ren, S., Zhou, Y., He, L.: Multi-object tracking with pre-classified detection. In:
Kim, J.-H., et al. (eds.) RiTA 2017. AISC, vol. 751, pp. 503–513. Springer, Cham
(2019). https://doi.org/10.1007/978-3-319-78452-6_40
13. Sarwar, F., Griffin, A., Periasamy, P., Portas, K., Law, J.: Detecting and counting
sheep with a convolutional neural network. In: 15th IEEE International Conference
on Advanced Video and Signal Based Surveillance (AVSS), pp. 1–6. IEEE (2018).
https://doi.org/10.1109/AVSS.2018.8639306
14. Sarwar, F., Griffin, A., Rehman, S.U., Pasang, T.: Towards detection of sheep onboard a UAV. arXiv preprint arXiv:2004.02758 (2020)
15. Wang, G., Wang, Y., Zhang, H., Gu, R., Hwang, J.N.: Exploit the connectivity: multi-object tracking with TrackletNet. In: Proceedings of the 27th ACM International Conference on Multimedia, pp. 482–490 (2019). https://doi.org/10.1145/3343031.3350853
16. Wang, Z., Zheng, L., Liu, Y., Wang, S.: Towards real-time multi-object tracking. arXiv preprint arXiv:1909.12605 (2019)
17. Welch, G., Bishop, G., et al.: An Introduction to the Kalman Filter (1995)
18. Yoon, K., Kim, D.Y., Yoon, Y.C., Jeon, M.: Data association for multi-object tracking via deep neural networks. Sensors 19(3), 559–673 (2019). https://doi.org/10.3390/s19030559
19. Zhang, Y., Wang, J., Yang, X.: Real-time vehicle detection and tracking in video based on faster R-CNN. J. Phys.: Conf. Ser. 887, 012068 (2017). https://doi.org/10.1088/1742-6596/887/1/012068
20. Zhou, S., Ke, M., Qiu, J., Wang, J.: A survey of multi-object video tracking algorithms. In: Abawajy, J., Choo, K.-R., Islam, R., Xu, Z., Atiquzzaman, M. (eds.) ATCI 2018. AISC, vol. 842, pp. 351–369. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-98776-7_38
21. Zulkifley, M.A., Moran, B.: Robust hierarchical multiple hypothesis tracker for multiple-object tracking. Exp. Syst. Appl. 39(16), 12319–12331 (2012). https://doi.org/10.1016/j.eswa.2012.03.004