Strathprints Institutional Repository

Dao, Trung-Kien and Le, Thi-Lan and Harle, David and Murray, Paul and Tachtatzis, Christos and Marshall, Stephen and Michie, Craig and Andonovic, Ivan (2015) Automatic cattle location tracking using image processing. In: 23rd European Signal Processing Conference, 2015 (EUSIPCO 2015), 2015-08-31 - 2015-09-04. ,

This version is available at http://strathprints.strath.ac.uk/53834/

AUTOMATIC CATTLE LOCATION TRACKING USING IMAGE PROCESSING

[2]Trung-Kien Dao, [2]Thi-Lan Le, [1]David Harle, [1]Paul Murray, [1]Christos Tachtatzis, [1]Stephen Marshall,
[1]Craig Michie, [1]Ivan Andonovic

[1] Dept. of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, G1 1XW, UK
[2] MICA Institute, Hanoi University of Science and Technology, Hanoi, Vietnam

ABSTRACT

Behavioural scientists track animal behaviour patterns through the construction of ethograms which detail the activities of cattle over time. To achieve this, scientists currently view video footage from multiple cameras located in and around a pen, which houses the animals, to extract their location and determine their activity. This is a time consuming, laborious task, which could be automated. In this paper we extend the well-known Real-Time Compressive Tracking algorithm to automatically determine the location of dairy and beef cows from multiple video cameras in the pen. Several optimisations are introduced to improve algorithm accuracy. An automatic approach for updating the bounding box which discourages the algorithm from learning the background is presented. We also dynamically weight the location estimates from multiple cameras using boosting to avoid errors introduced by occlusion and by the tracked animal moving in and out of the field of view.

Index Terms— Image and video processing; Object tracking in crowded environments; Cattle localisation;

1. INTRODUCTION

Agricultural resource management is critical in the farming of beef and dairy cows. In the past ten years, methods for automating the process of monitoring the behaviour of cattle have become increasingly important. As food demands grow, the average farm size has grown, and in the UK, this has almost doubled in the last 10 years [1]. As a direct consequence, farmers have significantly less time to observe their herd and are becoming increasingly reliant on technology. Automated decision support tools are now routinely offered to assist farmers to detect when livestock are in heat and hence exploit fertility [2]. These systems can also provide additional information such as mobility, eating patterns and posture and gate (standing/lying) information, which can be a strong indicator of animal welfare [3]. A combination of measurements on RF carrier signals (phase shift and signal strength) and acceleration data derived from a neck-mounted collar can be used to give a 3D location and posture fingerprint of the animal. This information can then be used to detect when an animal is in the vicinity of the feed lot and/or identify if it is resting for extended periods in the stall. It can also be exploited, for example, to estimate the amount of time an animal spends eating [3]. All of this information can subsequently be used to indicate livestock wellbeing.

While recent years have seen significant increase in the use of technology for the tasks described, behavioural scientists still manually view video footage captured from multiple cameras situated around the pens which house cattle in order to further understand their behaviour. We have therefore identified an opportunity to use image processing techniques to automate this analysis.

In this paper, we extend the well-known Real Time Compressive Tracking (RTCT) algorithm [4], to track the location of cows in video sequences over time. The RTCT algorithm has been evaluated in a number of application scenarios [4]. In most cases, RTCT outperforms alternative leading trackers including fragment [5], multiple-instance learning [6], semi-supervised tracker [7], online AdaBoost [8], ℓ_1 tracker [9], TLD tracker [10] and the Struck method [11] (see comparison in [4]). Due to its success and popularity, RTCT was recently extended for tracking humans in thermal imagery with high success rates [12]. Here, the authors explained a number of shortcomings found when applying the original RTCT technique to their data [12]. These include: the requirement to manually define a bounding box to contain the object of interest when this could be done automatically; and tracking errors and inaccuracies in complex scenarios e.g. when objects are occluded in the scene

The former is a simple initialisation problem. The latter is more complex and is a direct result of the learning process employed by RTCT. These issues are successfully addressed by the inclusion of a human detection algorithm which exploits specific properties of the thermal image data [12]. While the results are good and show improvement on the original RTCT approach, the same techniques cannot be directly applied to the problem at hand since the only video data available is captured using a standard RGB camera rather than a thermal one.

In this paper, we propose a novel cow detection algorithm to overcome the challenges described above. The proposed cow detector can be used to initialise the RTCT algorithm by providing an initial bounding box which contains the cow. We also demonstrate how the cow detection algorithm can be used to overcome the challenges posed by occlusion and to improve tracking accuracy in situations when a cow leaves and then later re-enters the scene. Furthermore, we provide a paradigm which allows
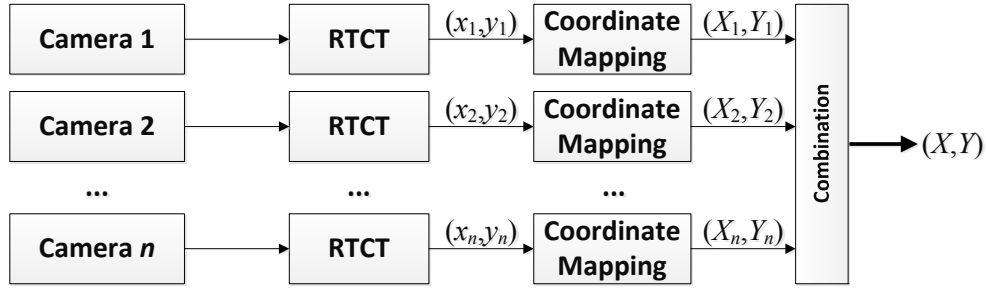
Fig. 1. Cow tracking using only RTCT

(a)   (b)

Fig. 2. RTCT example: (a) Cow localisation using RTCT, and (b) Cow localisation results at 314s using RTCT independently with two cameras

RTCT to be deployed on multiple video streams in parallel thus extending its application beyond single video streams as initially proposed [4].

2. REAL TIME COMPRESSIVE TRACKING

RTCT [4] requires a user to initialise tracking by providing a bounding box which contains the object of interest. In our case this would be the cow to be tracked. Once initialised, RTCT samples data from a number of rectangular regions within the bounding box and computes features from each sample region. This forms the "compressive" part of the algorithm and the features extracted from the current frame are used for matching in the next one.

When compressive feature extraction is completed, the next frame in the sequence is analysed by positioning multiple bounding boxes around the vicinity of the tracked object in the previous frame. Features are then extracted from each candidate bounding box, and a comparison of all features extracted from each box is performed with those stored from the previous frame. The position of the bounding box which minimises the distance between the extracted features from the current frame and the previous one is selected as the new object location. At this point, the bounding box and its associated compressed features are updated to contain the new location and the new set of compressed features for comparison in subsequent frames. The process then continues until all frames in the sequence have been analysed.

2.1. Application Issues in Cow Tracking

To evaluate the performance of RTCT for tracking a single cow in multiple cameras, an experiment was set up using the schema shown in Figure 1. It should be noted that RTCT was designed to process footage from a single cam-
era [4]. Hence, for our situation where n cameras being used to track a chosen cow, the video input from each camera is processed by an individual instance of RTCT.

The location of the cow in the video frame of the ith camera, denoted as $p_i(x_i, y_i)$, is the centre point of the bounding box computed by the instance of RTCT which is processing data from the camera in question. These image coordinates are then mapped to real-world coordinates which are denoted $P_i(X_i, Y_i)$. In this study, this mapping is implemented with a 2D bicubic interpolation on the basis of 9 pre-sampled points regularly distributed on a grid configuration. However, the accuracy could be improved with camera calibration methods [13,14]. Finally, the real-world coordinates from all cameras are combined to determine the final cow location $P(X, Y)$. This is achieved by simply computing the average position from all location estimates.

Despite the desirable advantages of RTCT such as low complexity and its high success rate compared to other algorithms in the literature [5-11], this implementation revealed several issues, including some like those previously encountered [12], that can prevent the cow from being accurately tracked. Firstly, we reiterate that RTCT has been developed for tracking an object in a single camera. Therefore, the localisation results from different cameras are obtained independently and the combination of them needs to be considered carefully in order to obtain more accurate results. Beyond this, there are several common situations when the tracked object will be easily lost, for example: when it moves out of view of the camera; when it is completely hidden or occluded by another animal; when its appearance changes quickly. Unfortunately, these situations happen very often in cow tracking; and this leads to a high loss rate when using the original RTCT algorithm.
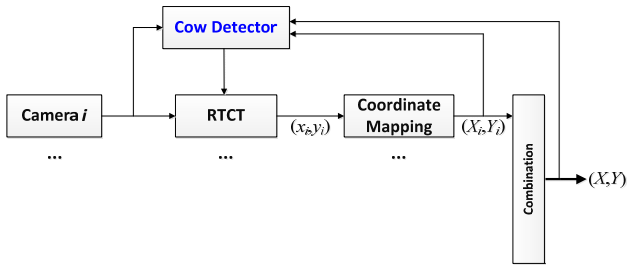
Fig. 4. Modified cow tracking with Cow Detector

To demonstrate this, Figure 2(a) shows the location outputs at different points in time obtained from tracking a black cow with two cameras using the described approach. Figure 2(a) shows the estimates obtained for X, Y, and D - the distance between the two output locations from the two cameras in real world coordinates when processing around 5 minutes of video footage. It can be observed that in the beginning, when the cow remains quite still, RTCT works well as D is relatively stable. However, when the cow starts moving at 287s, D increases very quickly. The cow moves away from the view of Camera 2 at 296s and is completely lost from the tracking in both cameras at 308s.

Figure 2(b) shows the tracking bounding boxes in the video frames of the two cameras at 314s which demonstrates that RTCT alone is not able to deal with more than the simplest situations in this application area.

3. PROPOSED SOLUTION

To address the issues described in the previous section, we have extended the original RTCT implementation to include an additional Cow Detector stage in each camera. This concept is illustrated in Figure 4. While it is anticipated that the addition of the cow-detection module will improve tracking accuracy, the extra computation required has an associated cost overhead in terms of processing time. For this reason, we also propose a method to detect when the location estimate from a camera is in error and hence when to apply the cow detection algorithms for correction. The main idea of the proposed solution is to update the tracking bounding box for the ith camera whenever its location estimate is detected to be erroneous. This offers an overall improvement in the tracking accuracy without expensive and, in many cases, redundant processing if the cow is being tracked with a high level of confidence. It is also shown in Section 3.3 how this solution can be used to detect the situation where a cow leaves the field of view and then re-enters at a later point in time.

3.1 Cow Detector Module

In this study, a Cow Detector has been implemented using the algorithm shown in Figure 3. The Colour Segmentation block is used to generate a grayscale image in which each pixel represents the similarity in colour of the original pixel to a predefined set of cow colours, say black, white and brown. The resulting image obtained by background subtraction and the colour-segmented images are combined by a weighted average. This combined image is then passed through a thresholding operator which generates a binary image where white pixels are likely to mark the
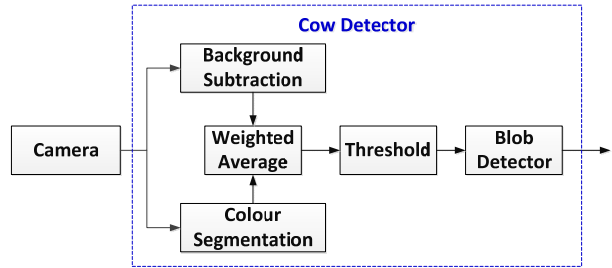

Fig. 3. Algorithm of Cow Detector

location of cows in the scene. The Blob Detector is then used to enumerate the blobs which satisfy the size and shape constraints imposed by a user.

The current implementation requires the manual setting of parameters which can be adjusted for optimal performance. For the Cow Detector module, the key variable parameters fix an upper and lower limit for the size of blobs which should be detected as cows. At present, the parameters should be set based on the field of view and the distance between the camera and the cow being tracked in the sequence. It is anticipated that when a sufficient number of videos have been processed these parameters can be fixed for each instance of RTCT in each camera.

3.2 Location Estimate Error Detection Module

Although the proposed cow detector is fast and simple, it still requires a small amount of processing and it is completely redundant to apply it in situations where a cow is being tracked accurately using RTCT. For this reason, we introduce an error-detection algorithm which is based on the distance d_i between the mapped cow location for that camera and the combined location. If d_i is higher than a predefined threshold value $d_{\max}$ then the tracking of the ith camera is determined as erroneous. Only in this situation where $d_i > d_{\max}$ should the cow detector be applied. Note that an appropriate value of $d_{\max}$ can be selected by a user to achieve the best results.

Figure 5 shows an illustrative example of the tracking with three cameras. Suppose that the tracking result obtained for Camera 3 is in error by a significant amount. This means that the distance between P_3 and P will be large, which in turn implies that $d_3 > d_{\max}$. In this example, the bounding box for Camera 3 is marked as erroneous and needs to be corrected. The correction is achieved using the Cow Detector described in Section 3.1 to detect all cows close to P within a $d_{\max}$ radius. Then, the cow which is closest to the position of the combined location P is used to update the bounding box position for Camera 3. This is shown in Figure 6 where the estimated P_3 has been updated as P_3'. It should be noted that in some situations the update can be unsuccessful in the case where no candidate is found by using the cow detector. In this situation the cow detector is applied repeatedly until: the cow is detected; or until such times that the end of the sequence is reached.
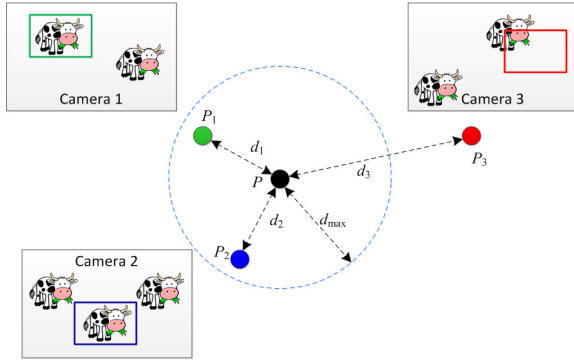
Fig. 5. RTCT tracking error detection

Fig. 6. Location correction using Cow Detector

3.3 Tracking Exclusion and Restoration

In cow tracking applications, it is quite common that the target will move out of the field of view of a camera only to reappear at some point later in time. When the cow moves away from a camera and out of sight, the tracking for that camera needs to be excluded from the process. In this study, a camera is excluded after a predefined consecutive number T_{max} times that the correction of the tracking bounding box for that camera fails. It is assumed that if this situation occurs, the cow has left the field of view of the camera in question. Note, however, that this situation could also arise if the performance of the Cow Detector is poor. In such cases, false exclusions due to poor Cow Detector performance can be overcome by using a higher value of T_{max} .

   Once a camera is excluded from the tracking, the Cow Detector is continuously requested in the following frames to detect if the cow comes back into view of that camera. This can be achieved by detecting a cow whose mapped location is close enough to the combined location P. That is, if the distance d_i between the estimated position of the detected cow and P is smaller than d_{max} . In situations where multiple cow candidates are found by the detector, the cow estimated to be closest to P is selected. The cow is completely lost from tracking when all cameras are excluded; that is, the cow does not appear in any video.

4. EXPERIMENTAL RESULTS

To test the improvement offered by the extensions of the RTCT algorithm proposed in this paper, the same video used to show the shortcomings of the original RTCT in Figure 2 was processed using the extended method described in Section 3. The goal is to track a cow living together with 15 others in a cattle shed with dimensions of 10m width and 15m long. The two cameras are mounted at the height of 3.5m. Table 1 provides a technical description of the video and a summary of the algorithm parameters used in this experiment. The parameters for this experiment have been determined empirically from the video. However, looking to the future, it is expected that when a sufficient number of videos have been processed by the proposed approach that a robust set of default parameters can be fixed in the routines for optimal performance.

   Figure 7 shows the mapped and combined location of the cow being tracked in the test sequence. It is clear that the addition of the Cow Detector significantly improves
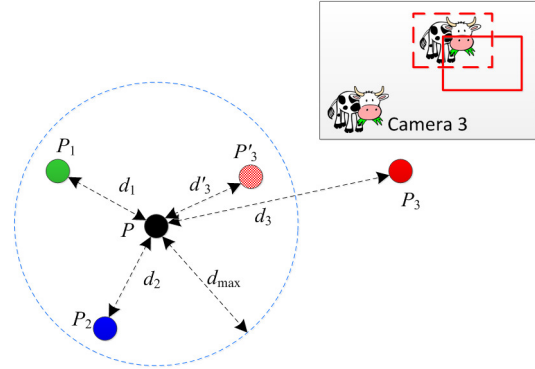
the results, and this time, the cow is tracked without being lost in all the 570s (9.5 mins) of video. This is in spite of the fact that that the cow moves out of sight of Camera 2 at 296s, then re-enters the scene at 319s. It then moves away again at 442s. For this reason, Camera 2 is excluded when the cow is out of sight but the cow is still tracked in Camera 1. The effects of the camera exclusion can be seen in Figure 7 in sections where there is no data. Clearly this exclusion also affects the computation of the distance, D. However, since the cow remains in view of Camera 1 throughout the sequence, the location results obtained for this camera can be used when data from Camera 2 is not available. As a result, the cow can be tracked throughout the 9.5 mins of video. From Figure 2, it can also be seen that D remains smaller than d_{max} almost all of the time.

Table 1. Parameter values

Parameter	Value
Video frame size	320×240 pixels
Video frame rate	27fps
Dropped per processed frames	1
Weight factor of colour-segmentation image over back-ground-subtraction image	3
Minimum blob size	500 pixels
Maximum blob size	10000 pixels
Distance threshold d_{max}	1.5m
Times of tracking failure for a camera to be excluded T_{max}	20

   Figure 8 shows the filtered (X, Y) cow location calculated as well as an estimate of the distance travelled by the cow throughout the sequence. These measurements have been extrapolated from the tracking data obtained using our extension of RTCT. On the basis of the variation of these data, it is possible to identify patterns that indicate times at which the cow is moving or others where it remains quite still. In the plots shown in Figure 8, areas which remain relatively flat indicate the cow is not moving. However, the sections of the plot highlighted between the dashed red lines mark areas of transition which indicate that the cow is moving at these times. Extracting information like this from the tracked results is of critical interest to behavioural scientists who are currently mapping out this process after physically viewing the videos. Furthermore, automatically extracting useful information like this
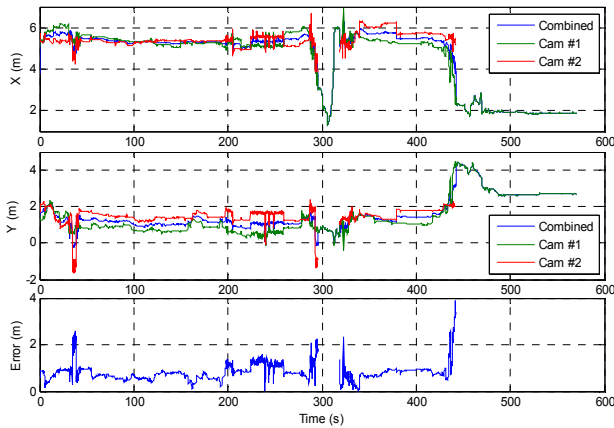
Fig. 7. Cow tracking results with Cow Detector

Fig. 8. Filtered cow location and traversed distance

will help in applications such as cow welfare optimisation, automatic annotation for cow surveillance, etc.

5. CONCLUSIONS

In this paper, novel extensions to the well-known RTCT algorithm [4] were introduced for automatic cattle location tracking. The proposed extensions allow RTCT to process video feeds from multiple cameras and to handle situations where cows are not simultaneously visible to all cameras. A simple cow-detector has also been implemented and added to the original RTCT routine. This automates the previously manual initialisation step of RTCT and, when used during tracking, makes the method more robust. In principle, the more cameras used in the proposed scheme, the more accurate the tracking performance.

For evaluation, 570s of footage from two cameras was used. The proposed approach was shown to successfully track the cow throughout the test sequence and was able to deal with the complexities introduced when a cow leaves and later re-enters the scene. Finally, the cow location estimates were processed to allow the traversed distance to be estimated and to help identify when the cow is moving or staying still. This forms the basis for extension into a number of applications to aid cow welfare optimisation and automatic annotation for cow surveillance.

REFERENCES

[1] Dairy Co, January 2013. "The structure of the gb dairy farming industry- what drives change?" Industry report, http://www.dairyco.org.uk/non_umbraco/download.aspx?media= 14129 [Accessed Jan. 14, 2015].

[2] NMR, 2012. "Silent Herdsman," [Online]. Available: http://www.nmr.co.uk/silentherdsman/ [Accessed Jan. 14, 2015].

[3] Lely, 2014. "Lely Qwes HR|Rumination activity," [Online]. Available:http://www.lely.com/en/milking/detection-system/qwes/rumination-activity [Accessed Jan. 14, 2015].

[4] K.H. Zhang, L. Zhang, and M.H. Yang. "Real-time compressive tracking," European Conference on Computer Vision, pp. 864-877, 2012.

[5] A. Adam, E., Rivlin, I., Shimshoni, "Robust fragments-based tracking using the integral histogram," IEEE Conference on Computer Vision and Pattern Recognition, pp. 798-805, 2006.

[6] B. Babenko, M.H., Yang, S., Belongie, "Robust object tracking with online multiple instance learning," IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 33, pp. 1619-1632, 2011.
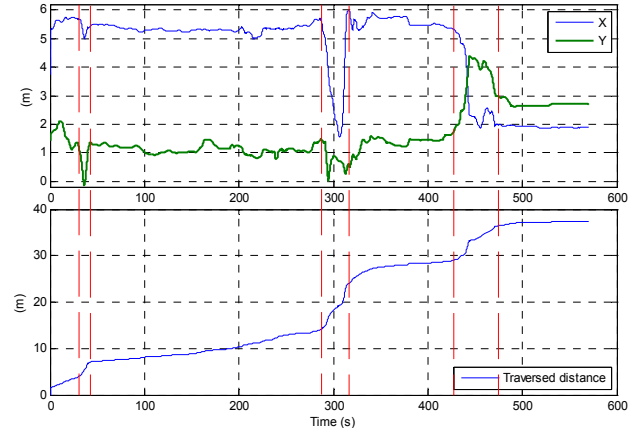
[7] H. Grabner, C., Leistner, H., Bischof, "Semi-supervised on-line boosting for robust tracking," Lecture Notes in Computer Science, vol. 5302, pp. 234-247, 2008.

[8] H. Grabner, M., Grabner, H., Bischof, "Real-time tracking via online boosting," British Machine Vision Conference, pp. 47-56, 2006.

[9] X. Mei, H., Ling, "Robust visual tracking and vehicle classification via sparse representation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, pp. 2259-2272, 2011.

[10] X. Kalal, J., Matas, K., Mikolajczyk, "P-n learning: bootstrapping binary classifier by structural constraints," IEEE Conference on Computer Vision and Pattern Recognition, pp. 49-56, 2010.

[11] S. Hare, A., Saffari, P., Torr, P, "Struck: structured output tracking with kernels," International Conference on Computer Vision, pp. 263-270 vol. 2011.

[12] Coutts, Fraser K., Stephen Marshall, and Paul Murray. "Human detection and tracking through temporal feature recognition," Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 22nd European. IEEE, 2014.

[13] Hartley, Richard, and Andrew Zisserman (2003), Multiple View Geometry in Computer Vision, Cambridge University Press, pp. 155-157. ISBN 0-521-54051-8.

[14] Zhang, Zhengyou, "A flexible new technique for camera calibration," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, pp. 1330-1334, 2000.