RESOLVING COMPLEX OCCLUSIONS OF OBJECTS DURING TRACKING USING REGION BASED SEGMENTATION

Abdullah Javaid Sindhu, Tim Morris
Department of Informatics
University of Manchester
P O Box 88, Manchester
a.sindhu@postgrad.umist.ac.uk, tim.morris@manchester.ac.uk
M601QD, UK

ABSTRACT
We have proposed novel techniques using region based segmentation for resolving complex occlusions of objects as they are tracked by a single, static video camera. The data is low resolution and noisy. The method improves the segmentation of moving objects obtained from any background subtraction method and then tracks objects continuously before, during and after occlusions. The method has been tested with real time complex occlusions of multiple people using a novel testing criterion that we have developed after critically analyzing all the possible scenarios of occlusions. The method has been proved to be successful with most of the testing criteria.

KEY WORDS
Occlusions, tracking, segmentations, 3D-histograms, testing criteria

1. Introduction

The ability of software to track a human or humans continuously during occlusions and also analyze their behaviour is a real need of today's security applications. If the software loses track of a human, then in reality, there is no possibility of being able to recognise the activity of a human, no matter how perfect the activity recognition is, as we may not be sure which particular person an activity or behaviour belongs to.

Therefore, real time human behaviour recognition is a procedure that must be preceded by successful tracking which becomes challenging when multiple people are in occlusion: a state where one or more tracked objects are hidden by fixed background objects or other tracked objects. During occlusion, a tracking system may not be able to correctly identify the occluded objects and it needs to re identify them when they become de-occluded: a state where a hidden occluded object reappears. Therefore, any tracking algorithm needs a set of identification features for matching objects before and after occlusions. Tracking and identification processes usually utilize a further set of processes for reasserting objects' identities, which are usually quite computationally expensive and their resource utilization increases with the number of targets in the scene. Therefore, for crowded scenes (such as supermarkets) where there is not only an increase in the number of people to be tracked but also the number of occlusions between them, real time tracking becomes difficult (due to limited processing power of the computer). So, it can be said that successful and real time behaviour recognition can only be achieved after successful and real time tracking, which can only be made possible if occlusions can be resolved in real time. We propose a novel method for resolving complex occlusions during real time tracking of multiple objects in crowded scenes based on improved region based segmentations.

The remainder of the paper includes a review of some of major current tracking algorithms. We then present a systematic analysis of occlusions which is used to evaluate the methods of resolving occlusions that are described in the section following. We then present sample results and conclude the paper with an analysis of the algorithm.

2. Background Review and Problems in the Tracking Systems

A typical tracking system normally has the following components,

- Detection of changes. This is normally achieved by comparing the current image with a model of the background. At its simplest, this is by subtracting the current image from an image known to be free of moving objects. Moving objects and any shadows they cast will be detected, unless they object is of the same colour as the background.
- Detection and removal of shadows. The differencing process will identify the changes that have occurred in the data: due to lighting changes, moved objects, and shadows cast by the objects. Shadows are identified by similar chromaticity but reduced intensity compared to the reference image.
- Identification of objects,

- Tracking objects and resolving occlusions.

The state of the art background subtraction algorithms [1, 2, 3, 4, 5] produce error prone results under the following conditions

1. The colour or grey value of the foreground moving object is similar to the background.
2. The foreground moving object is very close to the camera.
3. Slight camera oscillations produce ghost objects.
4. The moving objects occlude and de-occlude each other randomly.

After the background subtraction and shadow detection most tracking algorithms [9, 10] identify and label sets of contiguous pixels. These may correspond to complete objects; to disjoint portions of an object (if part of the object is similar to the background, it will not be recognised as object and could result in the object becoming disconnected); or to multiple objects that are sufficiently close to become merged [14].

If we know the likely sizes and shapes of objects, the discontinuous regions can be joined; regions smaller than the known size can be merged if they are in close proximity and the merged object is of an acceptable size. A range of sizes must be defined that will span the sizes of the objects projected onto the image plane from a range of distances.

There will be regions that were initially identified as corresponding to single objects that will merge into compound objects. In this case we do not assign all of the original labels to the region and do not attempt to divide the region into its constituent objects.

Many state of the art tracking algorithms [7, 9, 10, 11] use a Kalman filter of some flavour to predict an object's location at a future time using its previously observed movement. The filter provides a statistically rigorous method of maintaining a system model given noisy measurements.

Some systems [9, 10, 11] use colour histogram matching [8] to track and match multiple occluded objects. The problems with colour histogram matching are that firstly, it is computationally intensive. Secondly it is not possible to match the histogram of a compound object with the individual objects' histograms as the occluded objects are, by definition, occluded and information about them will be partially or completely unavailable. Despite these drawbacks, colour is a powerful feature that can contribute to identifying each and every object. We use the colour histogram matching technique [8] only in the case when objects become deoccluded and there is a guarantee that only one object is present in the said region. The technique is shown to work well, provided objects de-occlude singly.

3. EXPLANATION OF THE PROPOSED SYSTEM

3.1 Segmentation Improvement of Regions

Why is there a need to improve segmentation?

As we explained earlier, poor segmentation due to the similarity of the moving objects and the background, or the occlusion of an object with another object (static or moving) are the major obstacles to a successful tracking system as shown in figure 1, where four regions are detected but only one actual object is present.

To overcome this, we have proposed the following set of algorithms.

3.2 Region Merging

"Region Merging" is a process that uses the following simple rules to merge regions.

Rule 1: Regions whose sizes are smaller than the target object size are candidates for merging. The target object size is variable, depending on the range of the target.

Rule 2: Regions that lie within or close to the predicted area of a given target are candidates for merging. The tolerance on the predicted location varies according to the range of the target.

Rule 3: Regions may be merged if the merged region's size is consistent with the predefined region size.

The regions which satisfy the rules are merged, forming a complete logical object with acceptable accuracy. The rules successfully merge the regions corresponding to the given targets as shown in figures 2(c) and 3(c).

3.3 Region overlapping detection

We have found by experiment that the region merging produces some pairs of regions that overlap. For the clear assignment of regions to targets, this region overlapping should be resolved; figure 4, better illustrates the problem.

We use a "Region Overlap Detection" module to detect and separate these regions. Let R1(x1,y1,x2,y2) and R2(x1,y1,x2,y2) represent two regions where (x1,y1) and (x2,y2) represent their top left and bottom right coordinates. So, if the following test holds we can say that R1 overlaps R2.

$$\textit{Overlapped} = R1.x1 > R2.x1 \text{ AND } R1.x1 < R2.x2 \text{ AND } R1.y1 == R2.y1 \text{ AND } R1.y2 == R2.y2 \qquad Eq(1)$$

The regions belonging to different targets are then redefined based on these points

Figure 1(a):- A typical frame from a video clip.

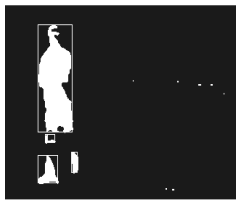

Figure 1(b):- The output of the region finding algorithm.

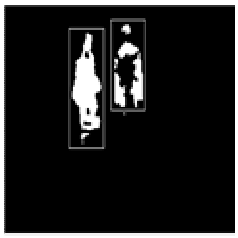

Figure 3(c):- The output of the region merging algorithm showing satisfactory segmentation.

Figure 2(a):- A frame from a typical video clip showing three moving targets

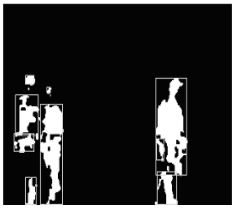

Figure 2(b): - The output of the region detection algorithm: the three targets have been separated into multiple regions.

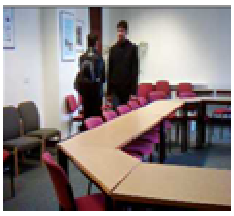

Figure 4(a): - A frame illustrating two un occluded, but close objects.

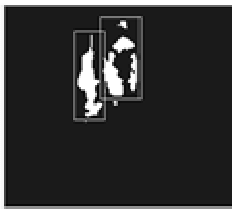

Figure 4(b):-The region merging algorithm's output: the two objects have been erroneously merged (overlapped).

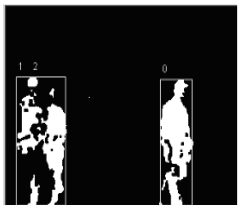

Figure 2(c):- The output of the region merging algorithm: the three targets have been resolved into one compound object and a simple object.

Figure 5(a): - A frame showing three targets.

Figure 5(b):- The region detection module has located one target (outlined in red) that is included within another.

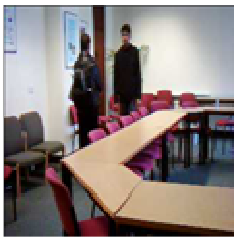

Figure 3(a):- A frame from a typical video clip, containing two moving objects.

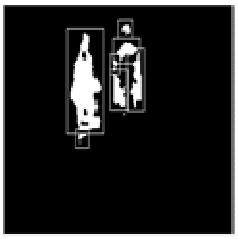

Figure 3(b): - The output from the region detection algorithm: the two objects have been split into multiple regions.

3.4 Included Regions Identification

Along with the problem of regions overlapping, there are also some regions included in other regions as a consequence of the region merging module, because some of the regions belonging to the centre of a single object can match with the background but outer regions may not. To overcome this problem, the module detects those regions and removes them from the outer regions (as we only need a single region to identify an object). The operation of the module is fairly straight forward: it first detects the smaller regions by measuring and comparing the area of the regions and then checks whether they are included by any other regions as follows. Let R1(x1,y1,x2,y2) and R2(x1,y1,x2,y2) represent two regions as before. If the following test holds then we can say that R1 is included in R2.

$$Included = R1.x1 > R2.x1 \text{ AND } R1.x2 < R2.x2 \text{ AND } R1.y1 > R1.y1 \text{ AND } R1.y2 < R2.y2 \qquad Eq(2)$$

The smaller regions are then removed from the outer regions.

3.5 Target Identification and Initialization

After obtaining a good segmentation of moving objects, the next task is to label the targets and initialize the tracking parameters. To identify targets, our system checks whether there is any region having a suitable width and height in the set of newly identified regions. We are attempting to track human targets so we are currently using 10 and 63 pixels as the width and height to be a minimum requirement for a region to be identified as a human target. These values depend upon the position of the camera, and the resolution of the image, so the system needs to be suitably trained with camera parameters to derive a reasonable size for the target. The values can be varied. If any likely target region persists for three successive frames, we create a new target in the database corresponding to that region and initialize it with the properties of the region. Some of the properties of the target initialization along with their explanation are given in Table I. The object's properties are calculated as soon as a new target is identified.

3.6 Tracking using a linear motion model

The purpose of this process is to update the targets' coordinate information, and any other coordinate dependent information, by searching the regions list from the latest frame and finding the closest match between the target and the region. The match is performed by comparing the predicted positions of the targets with the current regions' positions from the regions list. The target is identified with the region that lies closest to its predicted position. Finally the target's information is replaced with that of closest matched region.

3.7 Occlusion Detection

Occlusions are detected using the following two rules.

Rule1: The actual width of the target is significantly greater than the expected width.

Rule2: Two targets must be predicted to lie in the same area.

If these two rules are satisfied for at least two targets then it can be said that the targets are occluded. Rule 1 can be justified by considering the width of the region between two successive frames: it can only expand if two regions merge: a partial condition for the occlusion. For Rule 2, if the predicted positions of multiple targets are matched with an actual region obtained from the current frame, we have an occlusion. In occlusion, the occluded objects' identifications are assigned to the same region indicating a compound region, corresponding to multiple targets. The descriptions of the actual objects are stored in the object's database until the objects become de-occluded at which point the information is again updated and replaced.

3.8 De-occlusion Detection

It is necessary for a successful tracking algorithm to detect de-occlusion and to get information regarding the de-occluded objects. This event, which we term de-occlusion detection, is important as restoring the objects' identities is entirely dependent on it. We argue that this event can be recognised if these rules are satisfied.

Rule1: There must be a large reduction in the width of the region

Rule2: There must be some unassigned regions in the database of the detected objects after matching the predicted regions.

As can be seen in figures 6, 7and 8, the occlusion event is recognized as soon as the two rules defining an occluding event are satisfied and similarly for the de-occluding event.

4. Testing Criteria

After having performed a detailed analysis of occlusion events, we have formulated a set of test events. These are the occlusion and de-occlusion events that span the range of possible events. Any tracking algorithm should process these events correctly. The events are simply listed to enforce the vision for results in the next section.

A compound object (multiple people merged together) enters the field of view and de-occludes into

Event A.1: multiple simple objects or

Event A.2: multiple objects that subsequently occlude pre-existing objects.

Multiple simple objects enter the field of view and

Event B.1: occlude each other in a single event and then separate into simple objects,

Event B.2: occlude each other in a sequence of events and subsequently separate into simple objects,

Event B.3: occlude each other and then separate into compound objects,

Event B.4: occlude each other then the compound object subsequently demerges and merges with different combinations of components.

Occluded	*true/ false indicating that the target is in occlusion*
Total Occlusion	*an integer showing the total number of occluded targets contributing to the current object*
Current Frame	*current frame number of the region*
Minimum X	*the minimum X coordinate of the region*
Minimum Y	*the minimum Y coordinate of the region*
Maximum X	*the maximum X coordinate of the region*
Maximum Y	*the maximum Y coordinate of the region*
Centre X Predicted	*the predicted horizontal coordinate of the centre of the region*
Centre Y Predicted	*the predicted vertical coordinate of the centre of the region*
Centre X Previous	*the horizontal coordinate of the pervious centre position*
Centre Y Previous	*the vertical coordinate of the previous centre position*
Velocity X	*the horizontal component of the velocity of the object*
Velocity Y	*the vertical component of the velocity of the object*
Centre X	*the centre of region's horizontal coordinate*
Centre Y	*the centre of region's vertical coordinate*
Colour Histogram	*3D array containing the three dimensional colour histogram of the region*
Id	*an integer used to identify each region uniquely*
Need Correctness	*a flag showing the need to match parameters of the object to correct its identification*

Table I: The table indicates the list of properties of the newly identified target.

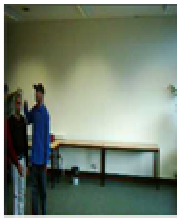

Figure 6(a):- An input frame; showing two objects.

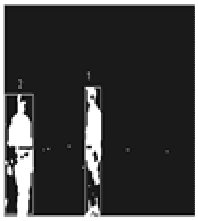

Figure 6(b):- The two objects and their identities prior to occlusion.

Figure 8(a):- An input frame; showing two objects.

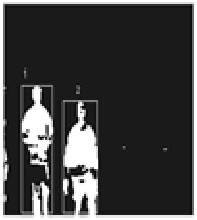

Figure 8(b):- The two objects and their identities after occlusion.

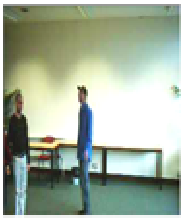

Figure 7(a):- An input frame; showing two objects.

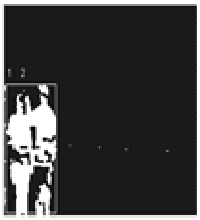

Figure 7(b):- The two objects and their identities during occlusion.

.

Figure 9: Showing a sequence of occluded and un-occluded objects being successfully tracked.

5. Results: Video Tracking Sequence of Multiple People during Occlusions

We have presented extracts from a video of the tracked objects as processed by our system. The objects become occluded and de-occluded several times, resulting in the combined tracking of the merged objects as shown by their respective identifiers adjacent to the bounding box. The results show that the system is sensitive enough to know when objects become occluded and de-occluded, an important feature of our system's robustness.

6. Evaluation of the Algorithm under the above Criteria

Our algorithm successfully meets the testing criteria A.1, and B.1, B.2 and B.3. The events A.2 and B.4 are not processed accurately by our algorithm. The reason is that when a compound object enters the field of view of a camera, we do not have any information showing the number of objects in that compound object and we do not have any way to initialize the properties of the individual objects. For example if three people enter the field of view of a camera and their side view is captured one object could be totally obscured by the others. So in this case, we only assign one identifier to the merged object and calculate all its collected properties. One way to solve this problem could be to count the number of objects in the compound object (possibly using a face detection or head counting algorithm), but still we have no guarantee that heads are visible as people's heads may be occluded. Even if the number of people in the region was known we would be unable to initialise the individual objects' properties (i.e. the colour histogram, width, height, face colour etc.). So, solving the A.2 event is a complex and challenging problem and we leave it as a future investigation. However, we have not as yet found any

system in the literature that is able to fulfil this testing criterion.

References :

[1] R. Cucchiara, M. Piccardi, and A. Prati, "Detecting moving objects, ghosts, and shadows in video streams," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25, pp. 1337-1342, Oct 2003.

[2] C.R.Wren, A. Azarbayejani, T.Darrell & A.Pentland, Pfinder: real-time tracking of the human body, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7), 1997, 780-785.

[3] Haritaolu, D. Harwood, & L. S. Davis, W-4: Real-time surveillance of people and their activities, *IEEE Transactions Pattern Analysis, and Machine Intelligence*, 22(8), 2000, 809-830.

[4] R. Collins, A. Lipton, T. Kanadeand, It. Fujiyoshi, D. Duggins & Y. Tsin, A system for video surveillance and monitoring, VSAM Final Report, Carnegie Mellon University, 2000.

[5] I.Haritaoglu, D.Harwood, & L.Davis, Hydra:Multiple people detection and tracking using silhouettes, *Proceedings of the Second IEEE Workshop on Visual Surveillance*, Fort Collins, USA,1999, 6-13.

[6] Y. Wu, T. Yu, G. Hua, Tracking appearances with occlusion, *Proc IEEE Conf on computer Vision and Pattern Recognition*, Madison, USA, 2003, 789-795.

[7] R. Cucchiara, C. Grana, M. Piccardi, & A. Prati, Detecting of moving objects, ghosts and shadows in video streams, *IEEE Trans Pattern Analysis Mach Intell* 25(10), 2003, 1337-1342.

[8] M.J.Swain, D.H.Ballard, Colour indexing, Int J *Computer Vision 7*(1), 1991, 11-32.

[9] S. J. McKenna et al., Tracking groups of people, *Computer Vision and Image Understanding, 80*(1), 2000, pp 42-56 2000.

[10] S.J. McKenna, Tracking colour objects using adaptive mixture models, *Image and Vision Computing, 17*(3-4), 1999, 225-231.

[11] Y.Raja, S.J. McKenna, S.Gong, Tracking and segmenting people in varying lighting conditions using colour, *Proc IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, Japan, 1998, 228-233.

[12] R.E. Kalman A new approach to linear filtering and prediction problems *Trans ASME – Journal of Basic Engineering 82*(D), 1960, 35-45.

[13] M. Isard, A. Blake, Condensation – conditional density propagation for visual tracking, *Int J Computer Vision 29*(1,) 1998, 5-23.

[14] R. Gonzalez and R. Woods, *Digital Image Processing*, Addison-Wesley Publishing Company,