A REGION BASED APPROACH TO TRACKING PEOPLE BEFORE, DURING AND AFTER OCCLUSIONS

Abdullah Javaid Sindhu, Tim Morris
Department of Computation
UMIST
PO Box 88
Manchester
M60 1QD
U.K.
a.sindhu, dtm@co.umist.ac.uk

ABSTRACT
We have proposed a novel approach to the tracking of multiple people before, during and after occlusions. The method is intended to deliver better segmentation from low resolution and noisy data using any background subtraction algorithm. A simple and novel tracking algorithm based on the region segmentation is presented. This can track clearly visible targets and occluded targets, even when the occlusion involves several people. Sample results are presented indicating the effectiveness of the algorithm. In addition we have analysed the occlusion scenarios and specified a set of test events.

KEY WORDS
Tracking, occlusion, histogram matching and computer vision.

1. Introduction

The ability to track an object is important in a range of applications ranging from security systems to behaviour analysis. Tracking is usually achieved by processing video data collected from a suitably located camera. The object or objects to be tracked are followed through the image sequence; the tracking may return image co-ordinates or a projection of the image co-ordinates into the world space. A frequent problem that is encountered is occlusion: when one or more of the tracked objects is hidden by fixed objects or other tracked objects, obviously the occlusion problem becomes more acute as the scene becomes more crowded.

The occlusion problem has often been “solved” by identifying and matching the participants before and after occlusion. However, this does completely solve the problem as no information is derived regarding the participants’ movements during occlusion, neither does it address the problem of resolving ambiguities when objects de-occlude. These are the problems that are addressed in this paper. We describe a system that is able to track multiple objects and resolve multiple, complex occlusions.

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

2. Review

Tracking of multiple people before during and after occlusion is vital for many applications and has been reported extensively in the literature. Most tracking systems [e.g. 1, 2, 3, 4, 5] have several common components:

- background subtraction to identify potential targets by comparison of an image with some model of the background scene,
- shadow detection to remove false positive pixels
- object detection: grouping contiguous pixels into blobs
- target identification and description: the derivation of characteristics of the blob which will enable the target to be recognised in succeeding frames
- background maintenance: the continual modification of the background model to adapt it to current lighting conditions, and
- tracking: following a target as it moves within the system’s field of view.

PFinder [1] builds a background model of the mean and variance of each pixel’s value. A Z-score could then be computed to decide whether a pixel was statistically different to the background. Wδ [2, 3, 4] used a similar technique, but approximated the variance by the range of values.

Shadows are usually removed by comparing the colour values of the shadowed region against the background: the colours will be similar and the intensity lower [6].
Targets are generally described by their colour distributions [7]. PFinder [1] used a combination of colour matching and a Kalman filter to track and verify objects. McKenna [8, 9, 10] has used colour information alone to track objects in complex scenes. Haritaoglu [4], however, used silhouette information to match objects, which would seem to be a less robust approach.

Whilst these authors claim to be able to track multiple objects, none of them seem to claim that they are able to deal correctly with occlusion. That is, they fail to reidentify correctly the objects that emerge when a group of targets separates. This is the problem that is the subject of this paper.

3. Analysis of Occlusions

Occlusion occurs when an object or objects that are being tracked is momentarily hidden by another object or objects. The object(s) causing the occlusion may be ones that are being tracked or may be fixed objects in the field of view. It is helpful to consider the various possibilities of how objects might occlude or be occluded in order to understand the difficulties that will be encountered when attempting to solve the occlusion problem.

We define a simple object to be a region of the image corresponding to one target and a compound object to be a region corresponding to multiple targets. We further define occlusion and de-occlusion events. The first is the combination of two or more objects to form a single, compound object, the second is the demerging of a compound object into two or more objects. Note that the compound object could be produced simply by the components touching each other – there might be no real occlusion. Finally we define entering and leaving events when an object enters or exits the field of view.

If the activity observed is to be analysed completely, we must create and maintain the mapping between targets and objects. Naturally, this is straightforward for simple objects, but less so for compound objects, especially when entering or de-occlusion events or multiple occlusion and de-occlusion events occur.

Consider an entering event. If it is a simple object, we may create the mapping between the object and the target. However, if it is a compound object, we must recognise the number of objects present and their descriptions before we can establish the mapping.

Occlusion events can be simply processed, assuming that the tracker is aware of the mappings between targets and the component objects. It is then a simple matter to reassign these targets to the newly formed compound object.

We may identify two types of de-occlusion events. In the first, the compound object splits into one or more simple objects, plus one compound object. In this case we ought to be able to identify the simple objects, the targets corresponding to the compound objects may then be inferred. In the second type of de-occlusion event multiple compound objects are produced: generally, there is insufficient evidence to assign targets to objects.

The problem is compounded when multiple occlusion and de-occlusion events occur. The simplest case concerns four people. If they enter the field of view separately, we may associate targets with simple regions. If pairs of objects occlude to give two compound objects, we may infer the mappings between objects and targets. If the two compound objects occlude we may still infer the object’s components. If this object were to de-occlude into two compound objects, then we will fail to identify the objects’ targets, all we are able to do is record a possible set of targets.

4. Testing Criteria

Given this analysis of occlusion events, we may formulate 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.

A compound object enters the field of view and de-occludes into

- multiple simple objects or
- multiple object that subsequently occlude pre-existing objects.

Multiple simple objects enter the field of view and

- occlude each other in a single event and then separate into simple objects,
- occlude each other in a sequence of events and subsequently separate into simple objects,
- occlude each other and then separate into compound objects,
- occlude each other and the compound objects subsequently merge and demerge with different combinations of components.

5. Description of Proposed Algorithm

Our tracking algorithm has the components common to all systems, as outlined above. We have used the background subtraction and shadow suppression steps of W' [2] and Sakbot [6] to identify targets. However, as figure 1 shows, this gives an imperfect subtraction with numerous false positive and false negative results. Portions of the target that resemble the background are not detected and shadowed regions are occasionally detected. This results in each target being detected as a number of disjoint regions, and some regions being merged that should not (e.g. the feet of the rightmost person in the figure). The problems are exacerbated if low
Figure 1a: a frame selected from an arbitrary sequence.

Figure 1b illustrates the regions derived by the $W^4$ algorithm: each target has been separated into a number of small regions because portions of the target resemble the background.

Figure 2a: a frame selected from an arbitrary sequence.

Figure 2b illustrates the regions derived by the $W^4$ algorithm followed by region merging. The separated regions have been successfully merged into the correct targets.

Figure 3a: a frame selected from an arbitrary sequence.

Figure 3b illustrates the regions derived by the $W^4$ algorithm followed by region merging. Targets 0 and 1 are correctly merged, it might be argued that target 3 has been incorrectly merged.

resolution or noisy data is used (as would be generated by a low cost webcam).

Morphological operators may correct some of these anomalies, but are expensive to implement. We have
taken a knowledge based approach. We are tracking people using cameras in standardised locations and can therefore predict the likely shape of targets in the images we capture.

5.1 Rule Based Target Identification

As shown in figure 1, people in the images we are processing appear to be of similar sizes. We could define the regions corresponding to people using a standard bounding rectangle whose height would be three to four times its width. We also observe that the fragments of a badly segmented person are close neighbours.

Targets are identified by applying a simple region merging rule: if two regions’ sizes are within certain tolerances, and the two regions relative separations are within certain tolerances, then the regions are merged. The tolerances that are used are related to the regions’ sizes, and are different for horizontally and vertically separated regions. Figure 2 illustrates a typical correct result of this process and figure 3 illustrates an incorrect result where two targets are sufficiently close that the region merging rule combines them. The system has also labelled all of the targets detected in these images.

Having identified targets in each frame of the sequence, the next stage of our algorithm will match each one against the targets currently being tracked. This tracking algorithm will also account for occlusions.

5.2 Region Tracking

Tracking systems typically use predictive algorithms (for example the Kalman filter [11] or the Condensation algorithm [5, 12]) to follow targets smoothly and efficiently. These algorithms maintain a model of how a target is moving and predict a target’s location at some future time. An image is captured at that time and the target is found. The error between the true and predicted locations is used to update the motion model. The algorithms fail to correctly follow targets if the motion is too irregular. The Kalman filter assumes constant acceleration or velocity whilst Condensation requires that the object is within some radius of its predicted location.

To avoid these problems, and recognising that people will not move in a predictable fashion, we do not use a motion model. Instead, we identify targets in each frame and match them with what was visible in the previous one. A very simple motion model is used, but only to prioritise candidates for the matching process.

A target is instantiated in the system’s database when it enters the field of view provided that its apparent size is consistent and it persists for four frames. The target is given a unique identifier (the numbers that appeared adjacent to each target in figures 2b and 3b). The target’s minimum bounding rectangle is computed and the width and height are recorded. We also compute the target’s velocity and colour histogram (using 32, 32 and 16 bins for the hue, saturation and intensity values).

The target’s velocity is used to compute an approximate location in each succeeding frame. The closest object found by the background subtraction is the first to be matched. The intersection of the regions’ colour histograms is used to match targets. The output images shown below include the subtracted region, the minimum bounding rectangle and the target identifier. Having matched the target, the database entry is updated with the new locations, rectangle size and velocity. The colour histogram is not updated.

Occlusion events are recognised from the targets’ predicted locations. A compound object is found if multiple targets are predicted to be located within the same segmented region in the new frame. In this case, the compound region is given the label of all the constituent targets, as in figure 3b. The original targets are maintained in the database so that they may be recognised when and if the object de-occludes. Size, velocity and component information is maintained for the compound object.

When a de-occlusion event occurs, the histograms of the new regions are computed and compared with the components’ so that the identifications may be restored. If a new region is not recognised, then it is likely to be a compound region itself. In this case, histogram matching is performed again and the best matches are used to identify the new region. In future, we shall attempt to use structural information to estimate the number of people in a compound region and thus the number of best matches we should seek.

6. Results

Intermediate results have been presented above. Four further sample results are presented in figure 4, which includes the original frame and the processed and labelled frame. It is important to note that these frames are derived from the same sequence; we would therefore expect that targets’ labels are maintained during and after occlusion events. This is found to occur.

7. Discussion and Conclusions

We have presented a system designed to track people moving with a fixed camera’s field of view. The system uses the W  and Sakbot algorithms to identify significant differences from a background image and to suppress the false positive results due to shadows. These two algorithms result in targets being detected as a set of neighbouring disjoint regions. Our system has implemented novel methods of combining these regions reliably into the original target. We have also implemented novel methods of following these regions and maintaining their identities during occlusion events: when two or more targets touch and merge or one hides another.
An analysis of the types of occlusion was presented, this has informed our design of the methods of treating occluding objects. It informed the description of a set of test scenarios of different sequences of merging and demerging events.

The system has been tested using low resolution and low quality data derived from a webcam. It has yielded accurate results, samples were presented.

In future, we shall investigate methods of estimating the number of people constituting a compound object. This will facilitate the mapping of targets to objects.

References:


Figure 4: The tracking system has segmented regions corresponding to people and given them a unique and consistent label that is maintained during and after occlusion events.