Detecting and Classifying Intruders in Image Sequences

Paul L. Rosin
Cognitive Systems Group,
School of Computing Science, Curtin University of Technology,
Perth, 6001, Western Australia

Tim Ellis
Machine Vision Group,
Centre for Information Engineering, City University,
London, EC1V 0HB, UK

Abstract

This paper describes a knowledge-based vision system for automating the interpretation of alarm events resulting from a perimeter intrusion detection system (PIDS). Moving blobs extracted over a sequence of digitised images are analysed to identify the cause of alarm. Alarm causes are modelled by a network of frames, and models are maintained for the scene. Due to poor spatial resolution, non-visual contextual information is required to supplement the image data. Probabilities are combined and propagated through the network by Subjective Bayesian Updating.

1 Introduction

This paper describes a knowledge-based vision system for analysing image sequences resulting from a perimeter intrusion detection system (PIDS) [2]. The PIDS contains a number of cameras viewing areas installed with a variety of alarms. When an alarm is triggered the image sequence spanning the alarm event is stored. The vision system's task is to interpret alarm events, discriminating between alarms triggered by human intruders and the many false alarms caused by animals, weather-related events, or noise. In addition, the false alarms should be sub-classified to enable the performance of the PIDS to be monitored. The analysis system must cope with the variations in natural illumination (changing position of the sun, cloud cover, shadows), as well as at night, when artificial illumination is low in contrast and uneven.

Tracking and recognising complex articulated objects in two-dimensional images of outdoor real world scenes contains several difficulties for machine vision. Problems include occlusion, shadows, and variations in lighting conditions. The alarm sources are not easily represented by geometric models because of the wide variety in shape of natural objects. Moreover, the articulation and flexibility of animals in motion and the changing viewpoint causes their appearance to vary considerably within a sequence. An additional problem in this application is that due to the large range of depth in the scene target objects often have a poor spatial resolution.

Given such incomplete and variable image data, a solution is found by supplementing the poor visual data by the non-visual data provided by the PIDS. This includes the current environmental conditions (e.g. weather, time of day, season), the types and locations of the triggered alarms (e.g. buried cables, vibration sensors), camera location, and image acquisition details.

The alarm classification system has been developed within a frame-based vision system called FABIUS [5, 6] which is implemented in Prolog. Frames provide a flexible and well structured representation for modelling the alarm sources. The pattern matching and backtracking facilities of Prolog make it well suited to designing control structures. Image processing algorithms, written in C for efficiency, are triggered from frames as demons. The disparate sources of data are combined and evaluated by Subjective Bayesian Updating [3] to classify the alarm sources.

2 Motion Detection

Motion is detected in the sequence of images acquired over an alarm event. Techniques such as optical flow or feature correspondence are inappropriate here since the interval between successive image frames is relatively long (up to a second) allowing the target object to move a considerable distance (several times its length) and substantially change shape.

Since the cameras are static we can use image differencing instead. This method has the advantage in that it is extremely sensitive to changing pixels between successive images or a reference image (e.g. motion) and is simple to implement in both hardware and software; however, it is not a robust technique if the camera suffers from any movement. Successive frames in the image sequence are subtracted from a reference image depicting the scene in its undisturbed state. Ideally all non-zero pixels in the difference image represent motion. To overcome the effects of noise the difference images are thresholded using a dynamic global thresholder to extract significant blobs. More details are given in [4].

Although automatically acquired reference images are available, they are prone to artifacts due to temporary disturbances in the scene (e.g. animals passing through) and gradual changes (e.g. moving shadows). An alternative is to generate a reference image directly from the image sequence [1]. We have developed a temporal median filter for this purpose. Each pixel in the reference image is generated by reading the corresponding pixels at the same location in each image in the sequence, and choosing the median of the pixel values. As long as the objects move substantially within the image this technique works well.

A problem that occurs in heavy winds is camera shake during the sequence acquisition. For a non-static camera the images will not be in registration, causing the difference images to be very noisy. To prevent this, individual frames in the sequence are aligned with respect to the first frame. Cross-correlation is performed in the horizontal and vertical directions, using a pair of lines through the image which are correlated with two equivalent lines in the base image. Since an interlaced image is used the shift between fields is corrected by the same technique. This correction procedure need only be invoked in windy conditions, a situation which can be anticipated by examining

the associated data file for the sequence provided by the PIDS.

3 Sequence Formation

Following thresholding, a boundary-based feature extraction algorithm measures the size and position of the binary blobs in the image. The extracted image blobs arising from motion are assembled into consistent temporal sequences. This temporal filtering allows noise blobs, that do not form part of any sequence but only occur sporadically, to be eliminated. Consistent blob sequences are formed based on:

- their real world separation
- consistency of blob area
- continuity within a sizeable portion of the image sequence
- smoothness of sequence track

Currently sequences are formed by examining all combinations of blobs. For efficiency this is implemented using nested iterative loops in C rather than backtracking in Prolog. However, the problem is combinatorial and becomes time consuming if there are many blobs. Improvements are made by preprocessing to eliminate blobs. All blobs whose real-world area is completely outside the range of any of the models are removed. Also, very slow moving objects can be efficiently detected, stored as sequences, and then eliminated from the set of blobs to be processed for faster moving blobs.

4 Scene Models

Since the cameras are fixed, scene models can be constructed for each individual camera position. These will be useful for augmenting the blobs' simple feature descriptions with additional image-based information. Three models were made:

- A map of the areas covered by the various alarm sensors
- A map labelling areas such as ground, fence and sky in the scene
- A camera calibration model

The alarm map enables sequences to be ignored if they do not intersect the triggered alarm zone. The semantic labelling of the image aids model matching by restricting the interpretations of blobs based on their location. For example, dogs do not appear in the sky, and rabbits do not climb up fences. The camera calibration enables range measurements to be made on the objects and for pixel measurements to be scaled into real world units on the assumption that objects touch the ground plane or some other modelled surface. (Note: This assumption is invalid for birds in flight. In this case, we tend to overestimate the distance of the bird from the camera, producing an overestimate of the objects size. Similarly, we would tend to overestimate the speed of the object, though this is complicated if the bird is flying directly towards the camera.)

5 Object Models

Due to the large range of depth, small target objects such as birds can have a very poor spatial resolution in the image, containing as few as 5x5 pixels. At night even larger objects will be poorly defined since the floodlighting tends to saturate the cameras. Given such poor spatial and/or intensity resolution only an approximate silhouette with very little internal detail can be distinguished. Objects are therefore represented by simple image-based features for appearance (e.g. projected area) and movement patterns (e.g. maximum velocities and accelerations). Lacking the image resolution to detect structural information, we must accept large error ranges for the feature parameters, since object shapes and sizes vary considerably depending on changing viewpoint and articulation of subparts. This causes some overlap of object model parameters, making reliable distinctions between similar objects difficult. Including behaviour patterns (e.g. birds tend not fly in heavy rain or wind; some animals are more active at night than by day) helps disambiguate model matches.

Objects are represented by frames. Features of a frame are described by slots, and the functional aspects of the slot are described by facets. For instance, the relative importance of the presence or absence of each slot is specified by a pair of weighting values attached as *weight* facets to the slot. Facilities such as data value restrictions, value defaults, and demons are implemented by other facets. See [5, 6] for more detail.

Each object model is partitioned into two components each represented by a frame. The first describes the characteristics of the individual instances of the animals. The second describes the dynamic behaviour of the animals over a sequence. An example of part of an object model is shown in figure 1.

frame fox			
	ako	value	animal
	scaled_area	weight	[1,5]
		pdf	[band,0.06,0.1,0.3,0.35]
	location	weight	[3,10]
		one_of	[ground,trees]
frame fox_sequence			
	ako	value	sequence
	sequence_of	value	fox
	speed	weight	[1,5]
		pdf	[band,0.0,0.0,8.0,15.0]
	acceleration	weight	[1,5]
		pdf	[band,0.0,0.0,0.5,0.6]
	time_of_day	weight	[1,1]
		pdf	[band2,600,800,1600,1800]
	wind_speed	weight	[1,1]
		pdf	[downslope,10,20]

Figure 1. Model frames for an alarm source.

A model taxonomy is built up using the *ako* link, facilitating property inheritance. The frame network is shown in figure 2. Alarm causes are divided into three major classes - human, animals, and other false alarms. The animal

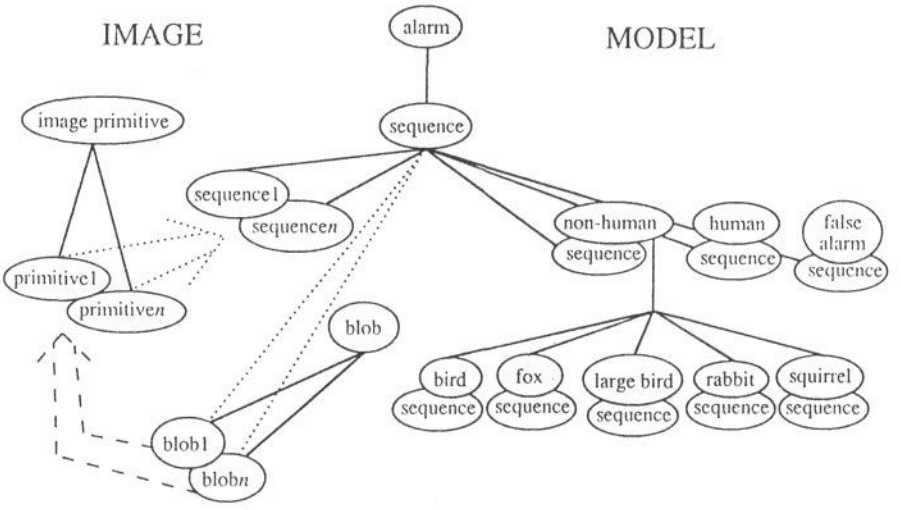

Figure 2. Frame network containing models and image data. Continuous lines represent ipo links, dashed lines represent instantiation links.

class contains sub-classes associated with more specific animal types. The general sequence model specifies that it is made up of several blobs, and contains various demons for calculating properties of a sequence such as velocity and acceleration. As can be seen in the figure image data is also stored in the network. Frames are created, and filled with the measured feature data, for each blob extracted and for each sequence formed.

6 Model Matching

Sequences of image blobs are matched against the set of models associated with the alarm causes, and classified as the best matching model. Model selection is performed top-down. The model tree is traversed a specified depth from the topmost alarm node, subclassifying the current classification at each level. In this application a human/animal/other classification is made first. If required the animal classification can be followed by finer classifications into specific animals.

The matching procedure involves the instantiation of a model and an image sequence. The matches between each individual image blob that makes up the image sequence and the individual model frame are evaluated. Each blob match requires each of its feature matches to be evaluated by using the image blob's slot value to index the probability distribution function in the model frame's slot's *pdf* facet. These individual slot matches are combined using Subjective Bayesian Updating [3] to form an overall match for the frame. In a similar manner these are propagated up to the sequence model which is evaluated by combining the probabilities of its individual blob frames and the slots corresponding to sequence feature matches. At each level in the model taxonomy the model with the highest match probability is chosen.

7 Results

Figures 3-4 show the results of processing two image sequences each containing eight 512x512 images at approximately 0.5-1.0 second intervals. Figure 3a shows the binary motion detection image containing two birds (plus shadows) taking off, overlaid onto one of the images from the sequence. Figure 3b shows the four sets of tracks that are detected (i.e. both birds and shadows). These are correctly classified as birds.

Figure 4a shows the motion detection image of a person running across the scene. Due to only slight contrast changes between parts of the person's clothes and the tree shadow the detection image is broken up, as shown by the minimum bounding rectangles of the set of detected image blobs shown in figure 4b. The sequence detector selects the most consistent set of blobs to form a track, as shown in figures 4c and 4d, and correctly classifies it as a human. The problem of fragmentation can be minimised by applying dilation and erosion operators to the binary image of detected blobs. In this example, a single erosion and dilation was sufficient to prevent breakup.

Table 1 shows the result of applying the classification procedure to 82 image sequences, mainly containing human subjects. As can be seen, the classification is not robust for subclassifying the false alarms, but does reliably detect human events. This is further emphasised in table 2, which groups all false alarm causes into a single class, and demonstrates the feasibility of the method for identifying the principal objects of interest - the human-generated alarms.

classification \ real identity	human	rabbit	pheasant	fox	bird	noclass
human	52	0	0	1	0	0
rabbit	0	4	0	1	0	1
pheasant	0	1	3	0	1	0
fox	1	1	0	2	1	0
bird	0	1	2	0	3	1
noclass	0	0	0	0	0	6

Table 1. Results of image sequence classification.

	human	false
human	52	1
false	1	28

Table 2. Results of classifying image sequences as human/non-human.

8 Conclusions

This paper describes a system for interpreting alarm sources from image sequences. The image-based data can be very poor, making a purely visual analysis difficult and unreliable. This is overcome by also utilising the non-visual data such as environmental conditions, types of alarm, etc. which are available to the system. Models and data are stored in a frame-based system which provides the facilities of model taxonomies, property inheritance, demons, and the

Figure 3a. Binary detected objects overlaid onto original image showing two birds (and shadows).

Figure 3b. Results of sequence detection for four tracks.

Figure 4a. Detected objects for a person running across the screen.

Figure 4b. MBRs of the set of image blobs detected.

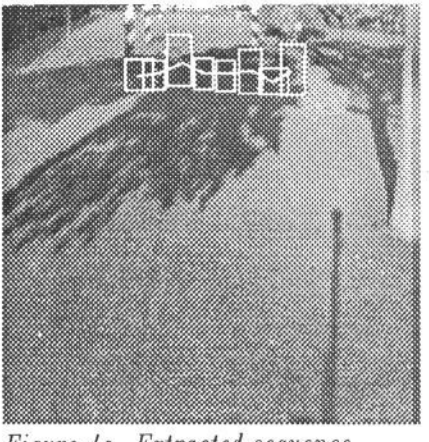

Figure 4c. Extracted sequence.

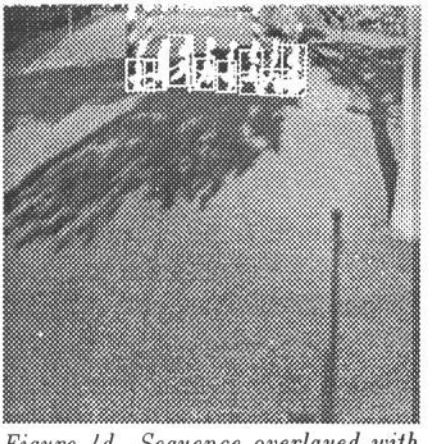

Figure 4d. Sequence overlayed with image blobs.

probabilistic combination of the different sources of data. The system performs robustly on the current test data of several hundred image sequences, correctly classifying human intruders under a wide range of illumination conditions, undertaking a range of activities (crossing alarm zones, crawling, climbing fences, etc.).

9 Acknowledgements

Dr. Rosin acknowledges the support and help of staff of the Home Office (Scientific Research and Development Branch) in this project. Dr. Ellis acknowledges the support of the UK Science and Engineering Research Council.

References

[1] S. Brofferio, L. Carnimeo, D. Comunale, G. Mastronardi, "A Background Updating Algorithm for Moving Object Scenes". In: Cappellini V. (ed) **Time-Varying Image Processing and Moving Object Recognition 2 (Proc. 3rd. Workshop, Italy)**, Elsevier, Amsterdam, 1990.

[2] N. Custance, P. Golton, T.J. Ellis, P. Rosin, P. Moukas. "The Design, Development and Implementation of an Imaging System for the Automatic Alarm Interpretation using IKBS Techniques". **Proc. Int. Carnahan Conf. on Security Technology**, 223-228, 1989.

[3] R.O Duda, P.E. Hart, N.J. Nilsson. "Subjective Bayesian Methods for Rule-Based Inference Systems". **Proc. Nat. Comp. Conf. (AFIPS Conf. Proc.)**, 45:1075-1082, 1976.

[4] T.J. Ellis, P. Rosin, P. Golton, "Model-Based Vision for Automatic Alarm Interpretation". **IEEE Aerospace and Electronic Systems Magazine**, 6:3, March 1991.

[5] P. Rosin, "Model Driven Image Understanding: A Frame-Based Approach". Ph.D. thesis, City University, London, 1988.

[6] P.L. Rosin, T.J. Ellis, "A Frame-Based System for Image Interpretation.". **Image and Vision Computing**, forthcoming, 1991.