Wavelet-based holistic sequence descriptor for generating video summaries

Andrew Graves and Shaogang Gong
Department of Computer Science
Queen Mary, University of London
London, England, E1 4NS
{andrew,sgg}@dcs.qmul.ac.uk

Abstract

We propose a video representation, the Video Scene Trajectory, computed from localised temporal-change in order to capture the holistic action content in a sequence. Such a representation is critical for action based genres such as surveillance. We show that an analysis of the trajectory shape is able to produce a temporal segmentation. Furthermore, we build an action based video summary using the Top Discriminative Active Pixels in the segments. Experiments are presented on real-world outdoor surveillance scenes.

1 Introduction

It is highly desirable to perform automatic video analysis using the video action content rather than static visual features. This commonly requires the detection of known events and activities in a specific scene through supervised training. However, content recognition approaches are difficult because of tracking problems in cluttered scenes and are limited because they are not scalable beyond the training scene. Our alternative philosophy is that video analysis can be achieved without the need to explicitly model the object-level content.

An important early task in video analysis is the formation of a *temporal segmentation*. It is concerned with dividing the sequence into segments that contain related content and providing a hierarchical decomposition. Many reported approaches find shot breaks at positions of sharp change between frames [14] and scene breaks by a process of shot grouping [7]. Techniques are usually colour-based and are often dependent on data being manually structured (i.e. edited into shots). There is a growing need for temporal segmentation approaches that operate on unstructured video data, for example on a corpus of home video [9, 6] or surveillance [5].

Generating video summaries is a key component of a video analysis system. A *video summary* aims to quickly impart knowledge to the viewer about the video content. Traditionally a video summary (or abstract) is a collection of static frames chosen to 'most' represent the video content and shown simultaneously. The process requires the detection of key-frames in the sequence and numerous approaches have been reported in literature, including using the first/last/mid frame or frame clusters [15]. Unfortunately a static frame based video summary does not provide any information on the action content that was present and is fundamental in action based genres like surveillance.

BMVC 2004 doi:10.5244/C.18.19

In this paper we develop a wavelet-based holistic video representation that can be used to solve the temporal segmentation task in unstructured video from a fixed view. We exploit activity information rather than colour because we found in previous work that colour information is not expressive enough in outdoor scenes [5]. In general, motions can be interpreted as orientations over time and analysed using motion-sensors [2] or image-slices [10]. Following the success of the temporal-change based approaches reported in literature [1, 4], we employ localised orientation filters to analyse the holistic appearance of temporal change in each frame. Similar recognition based methods have been reported for indoor activities [1] and complex outdoor events [4]. In this work, we use a holistic frame description because it requires no prior knowledge about the scene (such as object level detail) and is thus scalable to large video data-banks. This is critical if a diversity of surveillance videos with varying scene content needs to be analysed. We form a holistic video sequence descriptor by monitoring the change in local filter responses over time. Content breaks are found at points of discontinuity in the representation. We present a novel approach for video summarisation using the discriminative action that was present in each detected segment.

In Section 2 we describe our Video Scene Trajectory (VST) representation and demonstrate how it captures changes in the scene action content without the need for explicitly performing event and activity detection and recognition. In Section 3 we describe how the trajectory is analysed to produce: (1) a temporal segmentation using a trajectory approximation; and (2) a video summary using the most discriminative pixels in each segment. Our video summary approach is inspired by term weight computation in classical text document retrieval. In Section 4 we present results obtained from surveillance sequences and we conclude in Section 5.

2 Video representation without activity recognition

2.1 Holistic temporal filter based descriptors

We compute a frame-wise change description for reflecting activity in the scene. More precisely, the recursive computation of Pixel Change History (PCH) is defined as:

$$PCH_{\alpha,\beta}(x,y,t) = \begin{cases} \min(PCH(x,y,t-1)+\alpha, 1) & D(x,y,t) > Th \\ \max(PCH(x,y,t-1)-\beta, 0) & otherwise \end{cases} \quad (1)$$

where α and β are the accumulation and decay factors, D is the frame difference function $|I(x,y,t)-I(x,y,t-1)|$ computed between the grey-scale of neighbouring frames each smoothed with a Gaussian filter. The PCH is between $[0,1]$ for each pixel position where a high value indicates that a period of sustained change has taken place. The threshold Th, accumulation factor α and smoothing filter are used to reduce the effect of sensory noise. The decay factor β determines the size of the activity memory. See Figure 1.

The PCH image is divided into a grid of equally sized cells each of which is then described using the Haar wavelet transform. The cell-size determines the granularity of the descriptors and is chosen according to the scene layout and computational limitations. Moments of Haar wavelet coefficients are known to be effective for texture analysis and provide a good compromise between computational complexity and effectiveness [13]. Comparable approaches such as Gaussian derivatives and Gabor wavelets offer little im-

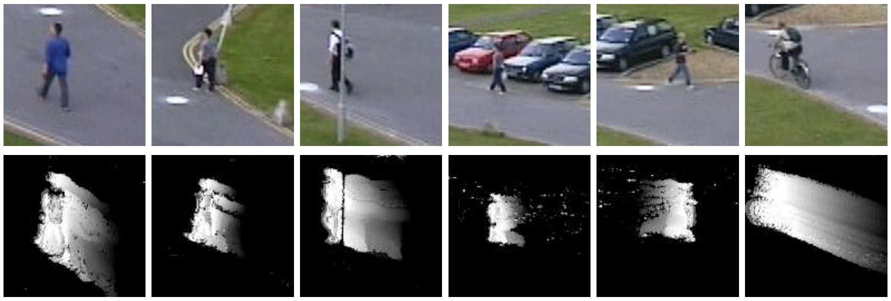


Figure 1: Extracts from an outdoor surveillance sequence and the computed PCH. Similar activities (person walking left) produce similar visual structures despite the cluttered background. A person walking right and a passing bicycle produce different PCH profiles.

provement in result and are known to be computationally demanding [11]. We therefore opt for the simple Haar basis because of our scalability demands:

$$\Psi(x) = \begin{cases} 1 & 0 \leq x < 0.5 \\ -1 & 0.5 \leq x < 1 \\ 0 & otherwise \end{cases} \tag{2}$$

that is applied using the standard wavelet transform:

$$\phi_i^j(x) = \phi(2^j x - i) \tag{3}$$

where x is the input to be translated using the number of scales j and position i. When computed upon the PCH space the transform captures the visual structure and directionality of the local action. We compute the mean coefficients in the LH, HL and HH bands, forming a 3D feature space that captures the amount of energy in the vertical, horizontal and diagonal directions.

We now form a frame description to capture the translation invariant action that is occurring. To this end, we perform clustering on the coefficient feature space computed for the whole sequence using k-means clustering. The cluster centroids each represent a commonly occurring class of filter response (or localised action shape). The classes form an *iconic vocabulary* used to describe each frame by (a) labelling each cell using the closest class by Euclidean distance, and (b) forming a histogram to capture the class occurrence in the frame. At each frame the number of class occurrences is constant:

$$\forall_{t=1}^{T} \left(\sum_{k=1}^{\kappa} FrmOcc(t,k) = N \right) \tag{4}$$

where *FrmOcc* gives the number of occurrences for a frame/class, N is the number of cells in the frame, T is the number of frames and κ is the number of clusters.

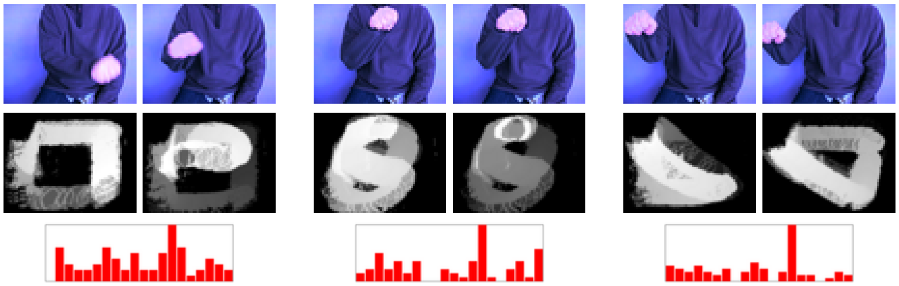


Figure 2: The frames and PCH computed from a sequence that captures a hand performing free drawing of three distinct shapes. The motion patterns of the shapes are clearly visible and are distinct. Each scene is characterised as having a different profile (the sum of frame histograms) of filter responses throughout its period.

2.2 A continuous scene descriptor

The key to our approach is that a scene can be defined as having a *similar profile of filter responses* throughout its period. This is illustrated in Figure 2. In order to form a continuous representation for the sequence that captures the long term content and thematic changes, we form a scaled cumulative histogram using the frame descriptions:

$$\forall_{t=1}^{T}\forall_{k=1}^{\kappa}\, ClsOcc(t,k) = \frac{ClsOcc(t-1,k)+FrmOcc(t,k)}{ClsOcc(T,k)} \quad (5)$$

where $ClsOcc$ gives the cumulative total at frame t for class k and $ClsOcc(0,k)=0$ for all classes. $FrmOcc$ is the frame histogram defined in Equation (4). It is clear that $ClsOcc$ increases monotonically with t for each class. The result is scaled between $[0,1]$ at each point using the final value at T. The use of a scaled cumulative histogram has the bonus effect of reducing the influence of noisy filters that occur very frequently.

The variations in the scaled cumulative histogram captures the filter activation combinations that describe the changing video content. The dimensionality of the histogram is equal to the number of clusters κ used to form the frame descriptions. This can be high meaning that it is difficult to analyse effectively and find the important profile changes. In addition many of the classes are in fact unimportant as they capture non discriminative action, e.g. very frequent background noise. Therefore, in order to focus the representation on what is important we computed the principal subspace of the scaled cumulative histogram using Principal Components Analysis (PCA) and re-project each t into the low dimensional eigenspace.

We use the first ω eigenvectors to form a Video Scene Trajectory (VST) for the sequence. The smooth/continuous trajectory indicates that the underlying filter response profile is approximately constant and the scene action content is not changing. For visualisation purposes we use $\omega=3$ in Figure 3, where (a) we show a trajectory produced using the method above but using a colour histogram as the frame description; (b) we show our action based Video Scene Trajectory. It is clear that using colour produces a very noisy trajectory that is not consistent with the video content whereas the VST has three distinct phases that correctly correspond to the content shown in Figure 2.

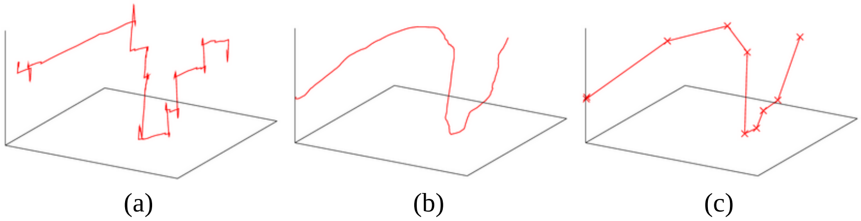


Figure 3: Trajectories for the sequence shown in Figure 2: (a) Computed using a colour histogram as the frame description. The lack of temporal consistency in the representation leads to errors in the temporal segmentation; (b) A Video Scene Trajectory. It is clear that the trajectory contains three distinct phases that correctly correspond to the the three scenes in the video; and (c) The piecewise approximation of (b). The approximation retains the shape of the trajectory using the most important vertices.

3 Video segmentation and summarisation

Given a video sequence of T frames, $\{F_1, F_2, \ldots, F_T\}$, we now address how the VST is used to generate a temporal segmentation of N segments, $\{S_1, S_2, \ldots, S_N\}$, each beginning at frame $t = S_i^S$ and ending at $t = S_i^E$. Considering that the trajectory is smooth when the action content in the scene is stable, our approach is to detect the key trajectory alterations and use these positions as the breaks. To this end, we generate a linear piecewise approximation of the trajectory that retains the key vertices using the Discrete Curve Evolution (DCE) algorithm proposed by DeMenthon *et al* [3]:

1. The 'relevance' of each vertex on the trajectory is computed using:

   $$rel(t) = dist(t-1,t) + dist(t,t+1) - dist(t-1,t+1) \qquad (6)$$

   where *dist* is the Euclidean distance. The relevance score *rel* is low if the point can be removed from the trajectory without significantly increasing the reconstruction error.
2. The vertex with the last relevance is removed
3. Repeat until the required number of vertices $\lambda = (N+1)$ remain

The retained vertices are used as the break points in a temporal segmentation and the content between vertices is considered a video 'segment'. An example approximation is given in Figure 3 (c). The DCE process is both highly efficient and effective, however one problem is that it operates on the trajectory in batch (i.e. it needs the whole trajectory to find the approximation). Alternative on-line methods could be exploited [8].

We now describe how we use the temporal segmentation for generating an action-based video summary. Our approach is to generate a segment summary frame for each segment using active pixels to indicate where action took place in that segment. To find the set of active pixels $\mathcal{P}_i$ that best represent segment i we evaluate each pixel using: (1) how active the pixel is in the segment; and (2) how good the pixel is for describing a segment considering the sequence. The motivation for 2 is that we wish to use the pixels that are most discriminative, i.e. are best for describing the unique content in the segment and minimise noise. This is similar to the *tf-idf* (term frequency, inverse document frequency) term weight strategy in text document indexing [12].

For each pixel we compute its activity in segment *i* as:

$$A_i(\forall^x, \forall^y) = \frac{\sum_{t=S_i^S}^{(S_i^E-1)} PCH(x,y,t) > Th}{\max A_i} \tag{7}$$

where Th is a threshold to determine significance and $\max A_i$ is used to scale the result to $[0,1]$. For each pixel we compute its discrimination ability in the sequence as:

$$D(\forall^x, \forall^y) = \log\left(\frac{T}{\sum_{t=1}^{T} PCH(x,y,t) > Th}\right) \tag{8}$$

A score for each pixel is computed using $\forall^x \forall^y A_i(x,y) D(x,y)$ and the pixels with the highest τ% of scores are used to form $\mathcal{P}_i$ for segment *i*. These pixels are the Top Discriminative Active Pixels in the segment and provide an indication of the key action that occurred. A segment summary frame (SSF) is formed by merging $\mathcal{P}_i$ with the first frame S_i^S in order to provide visual context:

$$SSF_i(x,y) = \forall^x \forall^y \begin{cases} 255 & if\ (x,y) \in \mathcal{P}_i \\ \gamma S_i^s(x,y) & otherwise \end{cases} \tag{9}$$

where γ is a scalar between $[0,1]$. A video summary is computed by forming an SSF for all of the detected segments and presenting them simultaneously to the viewer.

4 Experiments

We performed experiments on temporal segmentation and video summarisation using the following surveillance sequences:

Name	(x,y,t)	Description
PETS	$(768,576,3064)$	An outdoor uncluttered car park scene
RAMP	$(320,240,11000)$	A busy aircraft docking ‘ramp’ scene
T4	$(320,240,6400)$	An airport access road with moving traffic
T2	$(320,240,5000)$	A busy airport set-down area

The PETS sequence contains an uncluttered outdoor scene that suffers from noisy pixels due to camera vibrations. The RAMP sequence can be considered as semi-structured as the order of activities in the aircraft docking scenario are known. The T4 and T2 sequences are unstructured. Example frames are shown in Figure 6. Clearly it is difficult to impart knowledge about the sequence action content using static frames. It is interesting to note that, although the frames are colour, little useful colour information exists.

We computed the Video Scene Trajectory representation (Section 2) for all the sequences using the same parameters of $\{\alpha = 100, \beta = 10, Th = 30\}$ for computing the activity of Equation (1) and $\{\kappa = 20, \omega = 3\}$ when forming the scene descriptor. The choice of κ was made to ensure that the iconic vocabulary was sufficient to adequately describe each frame. The choice of ω was made because the top 3 principal components captures almost all (99.8%) of the variance. Note that the top eigenvector captures the cumulative aspect of the descriptor. We next computed the automatic temporal segmentation and video summary (Section 3) for each sequence using the number of segments $\lambda = [8,10,10,10]$ and $\{\tau = 10\%, \gamma = 0.5\}$. We used $\lambda = 8$ for the PETS sequence in order to compare against manually identified segments.

In Figure 5 we show summaries computed for the first three segments/scenes in the PETS sequence using key-frames, segment summary frames (SSFs) computed using the most active pixels in the segment, and SSFs computed using the Top Discriminative Active Pixels in the segment. It is clear that the first or mid frame approach often used in literature does not convey information about the content. Using the most active pixels in the segment is an improvement, but suffers from noise due camera vibration that particularly effects pixels at which there are strong spatial edges, e.g. at building outlines. The discriminant factor D from Equation (8) visualised in Figure 5 (f) captures these areas and reduces their influence. The result is a clearer SSF as shown in Figure 5 (e).

Manual breaks were identified in the PETS sequence and are marked in Figure 4 (a). However we emphasise that manual segmentation of unstructured data is highly subjective. Video summaries were produced and are shown in Figure 7 and can be used to formulate an idea of the content of the segments. We have highlighted four interesting points in Figures 4 and 7 for analysis:

A the automatic break and manual break are not close. However, upon inspection we found that a wide number of points can be considered as 'correct'. This non-error highlighted the problems of using manual segmentations for evaluation purpose.

B a short automatic segment was discovered that was not manually marked. Upon inspection we found that the segment contained unique 'car reversing' content that produced a distinctive signature in the trajectory.

C a number of manual breaks are missed. Upon inspection we found that the automatic method had grouped a number of very similar actions: people walking and a bicycle moving across the view.

D in the RAMP sequence, a seemingly large amount of trajectory variation is not accounted for in the segmentation. Upon inspection we found that the automatic segmentation had correctly grouped a large number of related unloading activities into a single long scene. The trajectory alterations are never violent enough to warrant an automatic break.

For the semi-structured sequence (RAMP), the discovered segments and resultant video summary do follow the strict timetable of events that is known to occur: (1) empty bay; (2) both plane arrival and ramp attachment have occurred; (3) loading vehicle activity can be seen; (4) small vehicle activities; (5) passing plane; (6) passing and obstructed vehicle; (7) another passing plane; (8) ramp activity; (9) plane departure. The video summary provides important and useful orientation for a viewer that is familiar with the scene. A semi-automatic search process can now continue as the viewer *drills-down* towards the target frames using knowledge about the expected order of events.

For the unstructured sequences (T4 and T2), the summaries produced reflect the action content and can be used to instigate deeper searches, e.g. in T4 scene 8 we might wish to analyse the segment further to discover the cause of the unusual shape that is present in the centre of the segment summary (in fact it is caused by a car reversing the wrong way down the slip-road; a very interesting section of video). However, the result for T2 is not entirely clear. This is partly because there is no structured/semantic content in the sequence, i.e. there is no structure to find, but mainly due to incorrect choice of granularity for the sequence. An automatic method for choosing the number of breaks λ is an area for future investigation.

5 Conclusions

We have presented a novel holistic wavelet-based video representation that can be used to perform video analysis without the need for explicit object and activity detection and tracking. The representation is based upon temporal-change and so implicitly reflects the action content that is present in the video. We monitor holistic action in the video over long periods and hence perform *topic spotting* in video (without knowledge of the topics).

We demonstrated how our representation can be used for performing temporal segmentation and then video summarisation, both crucial for building video analysis systems. An unsupervised holistic analysis of large surveillances videos using low-level primitive features facilitates a semi-automatic search process. Future work will consider how we can automatically find and label repeating trajectory structures of particular interest. We will also consider how to automatically choose the parameters and find the optimal segment granularity.

References

[1] A.F. Bobick and J.W. Davis. The recogonition of human movement using temporal templates. *PAMI*, 23(3):257–267, March 2001.

[2] O. Chomat, J. Martin, and J.L. Crowley. A probabilistic sensor for the perception and recognition of activities. In *ECCV*, volume 1, pages 487–503, Dublin, Ireland, June 2000.

[3] D. DeMenthon, L.J. Latecki, A. Rosenfeld, and M.V. Stückelberg. Relevance ranking of video data using hidden markov model distances and polygon simplification. In *Advances in Visual Information Systems*, Lyon, November 2000.

[4] S. Gong and T. Xiang. Recognition of group activities using dynamic probabilistic networks. In *ICCV*, pages 742–749, Nice, France, October 2003.

[5] A.P. Graves and S. Gong. Spotting scene change for indexing surveillance video. In *BMVC*, pages 469–478, Norwich, England, September 2003.

[6] G. Iyengar and A.B. Lipman. Content-based browsing and editing of unstructured video. In *IEEE International Conference on Multimedia & Expo*, pages 159–162, 2000.

[7] J.R. Kender and B.L. Yeo. Video scene segmentation via continuous video coherence. In *CVPR*, pages 367–373, Santa Barbara, June 1998.

[8] E. Keogh, S. Chu, D. Hart, and M. Pazzani. An online algorithm for segmenting time series. In *International Conference on Data Mining*, pages 289–297, San Jose, California, 2001.

[9] W.Y. Ma and H.J. Zhang. An indexing and browsing system for home video. In *European Signal Processing conference*, Finland, September 2000.

[10] C.W. Ngo, T.C. Pong, and H.J. Zhang. Motion-based video representation for scene change detection. *IJCV*, 50(2):127–142, November 2002.

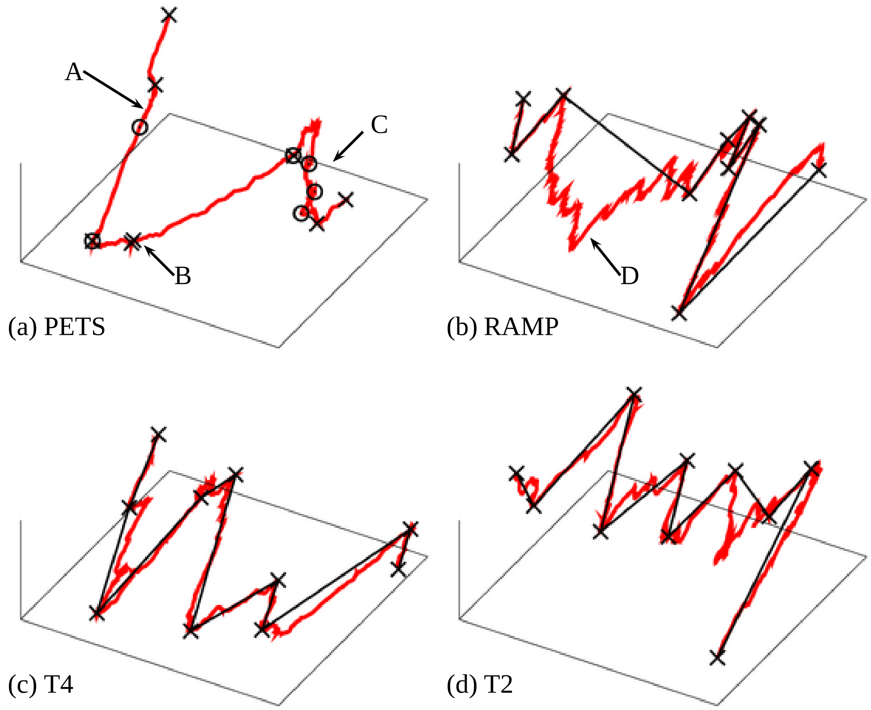
[11] M. Oren, C. Papageorgiou, P. Sinha, E. Osuna, and T. Poggio. Pedestrian detection using wavelet templates. In *CVPR*, pages 193–199, Puerto Rico, June 1997.

[12] C. J. Van Rijsbergen. *Information Retrieval*. Butterworths, 1979.

[13] M. Unser. Texture classification and segmentation using wavelet frames. *IEEE Transactions on Image Processing*, 4(11):1549–1560, November 1995.

[14] H. Zhang, A. Kankamhalli, and S. Smoliar. Automatic partitioning of full-motion video. *Multimedia Systems*, 1:10–28, 1993.

[15] Y. Zhuang, Y. Rui, T.S. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *ICIP*, volume 1, pages 866–870, Chicago, October 1998.



(a) PETS

(b) RAMP

(c) T4

(d) T2

Figure 4: The Video Scene Trajectories produced for the sequences shown in Figure 6. The automatic breaks marked by a cross and manual breaks with a circle (PETS only). Four interesting points are marked A, B, C and D as described in the text.

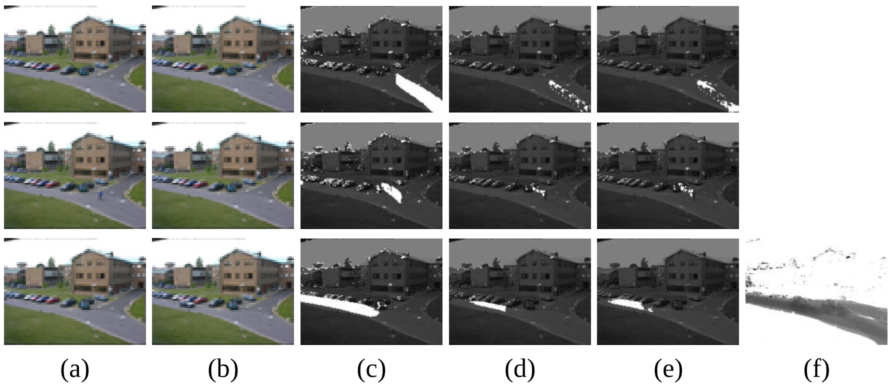


Figure 5: The first three automatically detected segments/scenes from the PETS sequence with alternative summaries: (a,b) the first and mid frame; (c) 100% of active pixels in the scene; (d) the top 25% most active pixels; (e) the top 25% discriminative pixels; (f) the discriminant D from Equation (8) used to compute (e). Our discriminative pixel approach reduces the effect of noisy pixels resulting in a much clearer segment summary frame.

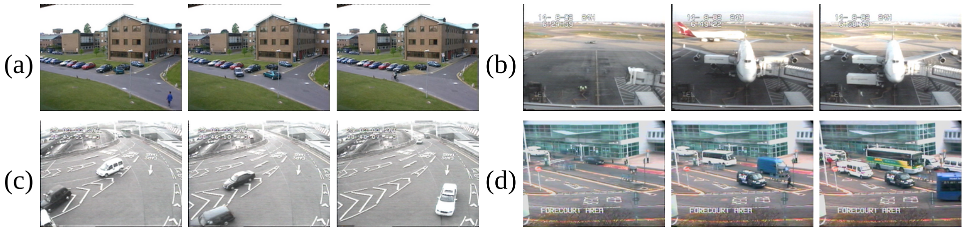


Figure 6: Illustrative frames from the sequences: (a) PETS; (b) RAMP; (c) T4; and (d) T2. It is clear that static frames convey very little information about the actual *content*.

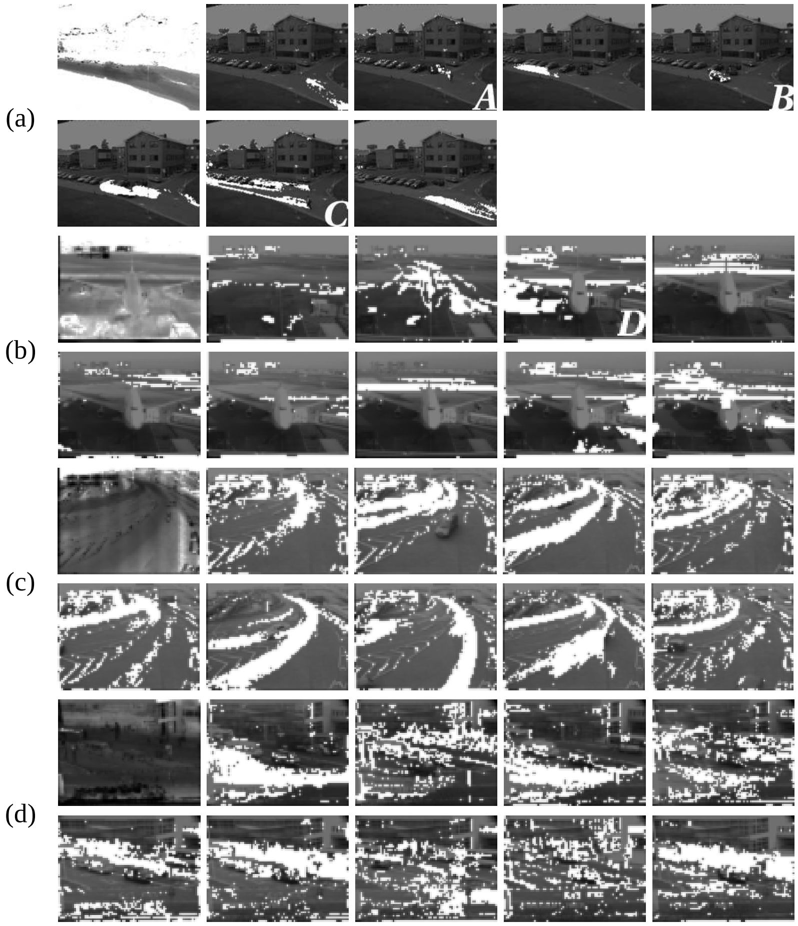


Figure 7: The video summaries produced for the sequences. In each case the pixel discriminant D from Equation (8) is shown first followed by the segment summary frames. Segments that correspond to points A, B, C and D in Figure 4 are marked.