Exploiting Temporal Discontinuities for Event detection and Manipulation in Video Streams[*]

Hugh Denman Erika Doyle Anil Kokaram Daire Lennon Rozenn Dahyot Ray Fuller
Departments of Electronic Engineering and Psychology
University of Dublin, Trinity College
Dublin 2, Ireland
hdenman@cantab.net edoyle4@tcd.ie

ABSTRACT

Discontinuities in any information bearing signal serve to represent much of the vital or interesting content in that signal. A sharp loud noise in a movie could be a gun, or something breaking. In sports like tennis, cricket or snooker/pool it would indicate a point scoring event. In both cases the discontinuity is likely to be semantically relevant without further inference being necessary, once a particular domain is adopted. This paper discusses the importance of temporal motion discontinuities in inferring events in visual media. Two particular application domains are considered: content based audio/video synchronisation and event spotting in observational Psychology.

Categories and Subject Descriptors

H.3.1 [**Information Systems**]: Content Analysis and Indexing—*Indexing Methods*

General Terms

Algorithms

Keywords

Video Retrieval, Motion Tracking, Information Retrieval, Event Spotting, Bayesian Inference

1. INTRODUCTION

Content based manipulation of an audiovisual media stream is only a defined problem once a domain of operation is acknowledged. It is true that unsupervised classification of feature space can help associate portions of a video stream with each other. However, it is only through the user that the semantic meaning and hence usefulness of that classification can be assessed or exploited. It is often the case that the act of acknowledging a particular domain of interest allows semantically relevant features to be identified and hence a more useful retrieval/access/manipulation system to be developed. Features that appear to generalise in their suitability to many domains, while at the same time directly semantically relevant, are therefore doubly important. Colour histograms are one such feature. It is well known that a discontinuity in the temporal evolution of image histograms between frames of a sequence are indicative of shot changes [1]. Shot changes are in turn the building block of semantic analysis [15, 3]. Camera motion is another such feature, being useful for identifying shot changes [2] as well as semantic events in cricket [9]. Object motion is clearly another universally useful and semantically relevant feature, although much more difficult to extract [14, 4].

This paper explores the use of object based motion cues for two aplications: content based audio/video synchronisation and event spotting in observational Psychology. The idea is to introduce implicit object based motion features that can be powerful when exploited in the right domain. The key to that exploitation is the detection of discontinuities in the temporal evolution of the features.

1.1 Detecting Percussive Movement

Consider a video of dancers dancing to a particular tune. At points which correspond to a beat in the original soundtrack, the dancers will typically exhibit a characteristic pose movement which demarcates a ‘phrase’ in the dance. Consider further that an editor wishes to create visual material to accompany a different music track, and has identified the dance footage as appropriate. A compelling way to bind the footage to the new audio material is to temporally warp the video such that frames which demarcate dance phrases are presented simultaneously with the beats of the new music track, causing the dancer to appear to dance to a different tune.

To achieve this automatically, it is necessary to locate instances in the video and audio track which should be aligned for a convincing final effect. Foreground/background segmentation using block based motion fields makes it possible to propose regions of foreground motion as primitive motion models. The motion models are then propagated over the visual sequence of the dancer. Discontinuites in the shape of the foreground mask, or in the evolution of a motion patch, are indicative of the end of a dancing *phrase*. These events are defined here as *percussive* movement events since there is a rapid change in shape or motion.

[*]Work supported in part by EU Project MUSCLE, EI Projects DysVideo, LastActionReplay and CASMS

MIR'05, November 10–11, 2005, Singapore.
Copyright 2005 ACM 1-59593-244-5/05/0011 ...$5.00.

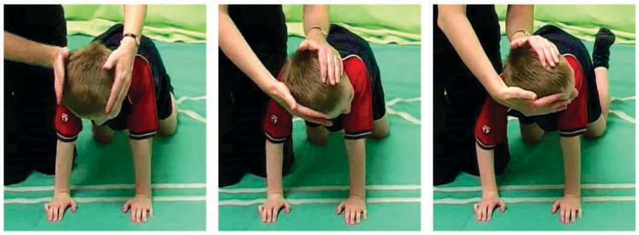

Figure 1: This example shows the movement designed to trigger the ATNR primary reflex (one of 4 reflexes being examined). It is called the Ayres test. In this test the child is on all fours. The experimenter rotates the head to the left and right slowly. The amount of movement made by the arms at the elbow during the rotation gives one clue about the severity of the retained reflex. In a non-dyslexic child, the hypothesis is that the elbow should not move.

1.2 Detecting episodes in Observational Psychology

McPhillips et al. [11] presented the notion that there is a quantifiable connection between Dyslexia and the retention of certain reflex movements. Dyslexia is now no longer seen solely as a problem generated by a higher-order brain malfunction, but as possibly a treatable disorder with a physiological rationale. Evidence was provided that in Dyslexics, certain *primary reflexes* [7] are retained. In subsequent development, these reflexes become integrated into postural reflexes to allow the child to progress to the next stage of movement. But in dyslexics, early reflexes may persist. The work of McPhillips et al. also indicates that Dyslexia can be treated by retraining the central nervous system by slowly repeating these movements. Hence the connection between the treatment of Dyslexia and a movement therapy.

The **DysVideo** [8] project at Trinity College was set up to observe the development of 400 children aged below 6 years. Each child is observed through 3 sessions of 20 minutes, each 6 months apart. The session is composed of 14 exercises that are designed to trigger each of four primary reflexes. For example, Figure 1 shows the movement designed to trigger the ATNR[5, 6] primary reflex. The experimenter rotates the head of a child right and left. While doing so any involuntary bend in the arms is noted. The idea is that the presence of that reflex is in some way correlated with the presence of Dyslexia.

There are two main problems. The first is the retrieval aspect: given up to one hour of recorded video per session, is it possible to automatically retrieve the two minutes of material in which a child is actually performing the exercise? The second is to assess during that portion of video how well the movement of the child matches some expected performence measure.

Previous published work by the authors [8] discussed preliminary system design and the start of tool development. The testers use a palmpilot to generate DTMF audio tones at the start and end of each recorded session that are recorded simultaneously onto one audio channel as the video recording proceeds. These DTMF tones encode numerical child ID codes as well as experiment type. Extraction of this information from the audio track is done automatically (FFT analysis) when the data is transferred from tape to disk later. This information yields a coarse parsing of the video material. The coarse parsing generally indexes a 20 second clip of material within which the motion to be measured occupies about 5 seconds. In each experiment however, the child undergoes some stylistic motion. Hence is it sensible that delineating the limb in question and tracking it through the 20 second clip, would yield a more accurate index for parsing. This is an example in which it is possible to exploit context much more heavily to achieve object extraction and hence tracking.

The essential idea is to generate temporally evolving motion features and detect the onset of events as discontinuities in the process itself. For instance, while the child is being explained the motion to be performed, there is a great deal of sometimes random activity. However, once the test begins, the activity is much more homogenous. This change in behaviour is used both to index the relevant portions of video as well as to instigate a tracker for the assessment of motion.

2. OBJECT DELINEATION

In both case studies outlined above the first step is to exploit context to delineate the object of interest and hence facilitate object tracking. In the case of Psychology (Psy), this is possible to a large extent because of the stylistic nature of experiments. However, in the Percussive Movement (PuM) study the only constraint on the content is that the foreground moving objects are of interest. This implies that the Psy example is able to exploit more feature information than in the PuM study.

2.1 Psy Body delineation

Head and arm localisation is facilitated by skin detection achieved with simple colour segmentation process. The requirement is to configure a label field $l(\mathbf{x})$ that is 1 at pixel sites $\mathbf{x}$ containing skin and 0 otherwise. The algorithm is as follows.

1. Candidate pixels expressing skin ($l(\mathbf{x}) = 1$) are detected by colour thresholding (from [13]) using the following criterion

$$l(\mathbf{x}) = 1 \text{ if } \begin{cases} (R > 95)\&(G > 80)\&(B > 40) \\ (R > G)\&(R > B) \\ (R - min(G, B)) > 10 \\ (R - G) > 15 \end{cases} \quad (1)$$

$$= 0 \quad \text{Otherwise} \quad (2)$$

   The various parameters used in delineating the colour region were determined from the lighting used in the pictures recorded. This is the same throughout 100 hours of recording. The first two criterion delineate skin colour, while the last two reject false alarms due to pixels that are near grey or near yellow.

2. The label field $l(\mathbf{x})$ is post-processed to smooth the surface. This is achieved using morphological closing with a dilation element of 3 pixels and an erosion element of 4 pixels.

As shown in figure 1 the arms are generally the largest area of skin exposed in the view. In addition they are near vertical. Hence a vertical sum (integration) of the label field

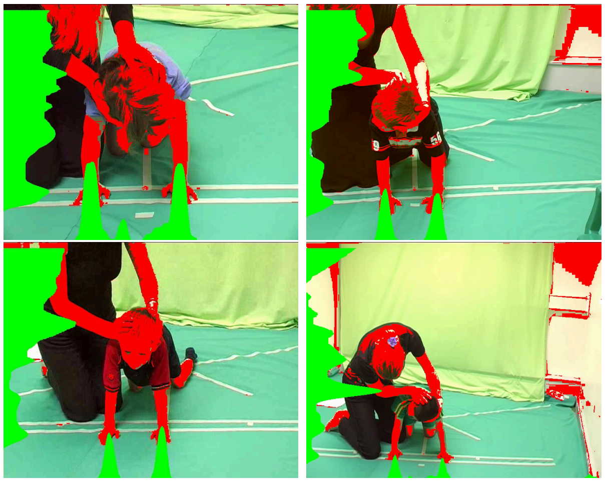

Figure 2: Example frames from different sequences showing the results of skin detection and hence body localisation. The detected skin pixels are coloured in red. The horizontal and vertical projections of the label field are shown in green along the left and bottom edges of each frame. This illustrates that the lobes in vertical projection correspond to arm location. The first mode in vertical projection (from the bottom of the frame) corresponds to arm location. The estimated lines delineating the child's arms, head and hands are shown superimposed on the image.

yields modes corresponding to the horizontal position of the arms. Given the detection field $l(\mathbf{x})$ the vertical projection is defined as $p^v[h] = \sum_k l(h,k)$. Noise in $p^v[h]$ is removed by filtering with a Gaussian filter with 9 taps and variance 1.5. To detect modes in $p^v[h]$ the two most significant maxima are selected that are at least 50 pixels apart. This allows robustness to false alarms within a single arm segment. Figure 2 clearly shows the correlation between lobes and horizontal arm location for 4 different recordings of 4 different children. Note the false alarms due to poorly detected skin in the background (due to strangely coloured walls) are rejected with this process.

Locating the hands is achieved through the horizontal projection of the label field $p^h[k] = \sum_h l(h,k)$. The first maxima corresponds roughly to the middle of the hand position because of the orientation of the child in the view. This is shown in Figure 2. The very first non-zero projection corresponds to the start of the hand location. From observation of body proportions over 10 hours of experiments it is possible to relate the distance between the start of the hand and the middle to the position of the wrist, D. The wrist location is hence taken to be $1.5D$. In addition, the average forearm length is approximately 2.5 the hand length in this view, hence the location of the elbow can be roughly delineated vertically. This enables a bounding box to be placed that contains the hand and arm locations. The process is found to be better than 99% accurate in these sequences, provided the child adopts the correct position. Typical results are shown in Figure 2. For video examples, see `www.sigmedia.tv/research/indexing/dyslexia/`.

The location of the arms is used to bound the head location horizontally. Therefore, head location is assumed to be contained within a column of the image bounded by the left and right arm locations. Unfortunately, detection of the head using projections is not reliable because in horizontal projection the face of the experimenter can often cause ambiguity. Instead a motion based strategy provides a solution to head localisation and rotation estimation simultaneously.

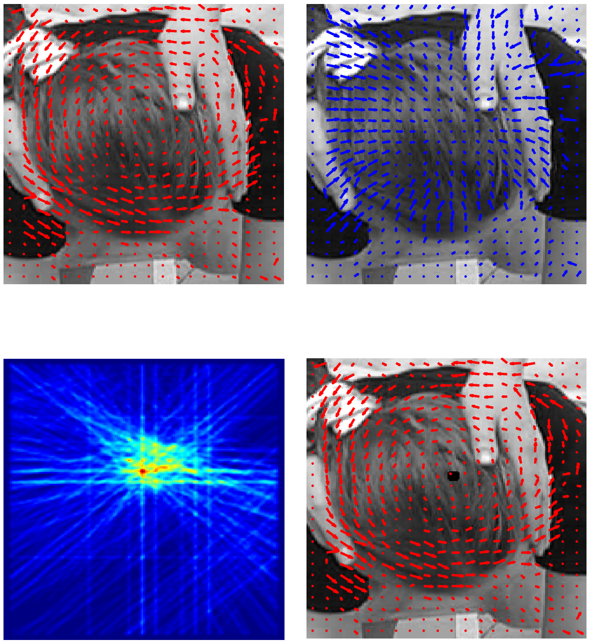

Figure 3: Estimating the centre of rotation. The process begins with block based motion estimation, the leftmost picture shows typical vectors superimposed on the image. Perpendicular vectors are then caculated (as shown in the next image), and these extended throughout the image and accumulated into an Accumulator array (third image from right). The mode of the accumulated vectors in the array yields an estimate of the head centre shown as the black dot in the rightmost image.

2.2 Head centre estimation

The centre of rotation of the head and the amount of that rotation are important features for parsing and measurement. Finding the centre of rotation of the head can be achieved without explicit head localisation. Figure 3 shows the result of estimating translational motion between two frames exhibiting head rotation. A multiresolution, gradient, block based technique was used (see Kokaram[10]) for motion estimation. Lines perpendicular to each motion vector, should intersect at the centre of rotation, provided rotation is occurring. When they do not intersect typically no rotation is occurring and hence the video material is irrelevant anyway. To estimate this centre of rotation therefore, lines perpendicular to the direction of motion in each block are accumulated in an accumulator array of the same size as the observed image. In practice of course the lines do not intersect exactly because of problems in sampling the lines during their entry into the accumulator array. Filtering with a Gaussian filter (anti-aliasing) of size 9×9 and variance 1 alleviates this problem (see Figure 3). A typical accumulator centre of rotation estimation process is shown in figure 3.

Detection of the maximum in the accumulator array yields an estimate for the centre of rotation. The height of a mode yields a confidence measure for the centre estimate. The technique is robust enough that when rotation is ongoing, the centre of rotation is noticeably stable. While this feature could yield a simple index to the useful video, estimating the rotation itself is more useful. This is discussed in the next section.

2.3 PuM Objects

Objects that move coherently are expected to belong to some semantically relevant portion. Unfortunately, the object that is undergoing the Percussive Motion may be perceivable by a human in the context of the performance (e.g. subtle movements of the hands) but not necessarily significant enough to be quantitatively assessed. Hence two kinds of object analysis are employed. A simple foreground segmentation process based principally on displaced frame difference, and another based on motion clustering.

In either case, global motion of the scene must be accounted for and this is estimated using the multiresolution gradient based scheme outlined in [12]. The sequence motion model is therefore as follows.

$$I_n(\mathbf{x}) = I_{n-1}(\mathbf{A}_g\mathbf{x} + \mathbf{d}_g + \mathbf{d}(\mathbf{x})) + e(\mathbf{x}) \qquad (3)$$

where the pixel intensity at position $\mathbf{x} = [i, j]$ in frame n is $I_n(\mathbf{x})$, the global motion of the scene has an affine component $\mathbf{A}_g$ and a translational component $\mathbf{d}_g$, and the local motion of the pixel at $\mathbf{x}$ is translational $\mathbf{d}(\mathbf{x})$. Hence $\mathbf{A}_g, \mathbf{d}_g$ are fixed over the whole frame, while the local motion $\mathbf{d}(\mathbf{x})$ would vary depending on the foreground motion. $e(\cdot)$ is a residual model error.

Regions undergoing foreground motion constitute a foreground map, $l^f(\mathbf{x})$. Pixels that are part of this map will be poorly matched with the previous frame if only the global estimate is used. Hence a simple mechanism for generating this map is to threshold the difference between globally motion compensated frames. Hence given $DFD_g = I_n(\mathbf{x}) - I_{n-1}(\mathbf{A}_g\mathbf{x} + \mathbf{d}_g)$, all pixels with $DFD_g > 15$ in this case are assigned to the foreground motion map. This map is smoothed with a morphologial filling operation uwing element size 5×5.

There is a huge amount of literature on motion based segmentation but the motion segmentation step is only one aspect of the PuM detection process. Hence the idea is to attempt to limit the computational complexity of the process by employing a simple local motion clustering step. Local motion is estimated using a a multiresolution, gradient, block based technique presented in Kokaram[10]. Local motion segmentation is therefore performed on a block basis (9×9 blocks in this case). Local motion estimation proceeds after global motion compensation, hence yielding an estimate for $\mathbf{d}_g(\mathbf{x})$ in the model above. Assuming that the first frame contains no local motion (the usual case in most performances), local motion exceeding a threshold of 3 pixels away from the background motion magnitude (i.e. the results of applying $\mathbf{A}$ yields blocks that are candidates for local motion clusters. Local motion blocks that are spatially adjacent are then assigned to a single cluster. These clusters then constitute the current local motion objects O_k^i, indicating the ith object in frame k. These are associated with some unique label, such that at a pixel $\mathbf{x}$ undergoing motion corresponding to the ith object, the label is $O_k^i(\mathbf{x})$.

The motion of each region with the same label $O_k^i(\mathbf{x})$ is modelled as a Gaussian with mean vector $\hat{\mathbf{d}}^i$ and covariance matrix $\mathbf{M}^i$ (a 2×2 matrix). Gradients in the blocks are used to weight the calculation of mean and covariance. The task is to assign blocks in the next frame to particular motion models, or to an outlier class Z. The outlier class allows the instantiatiation of new moving objects. Class assignment is based on a MAP criterion. Given site $\mathbf{x}$ it is required to manipulate $p(O_k^i(\mathbf{x})|\mathbf{d}(\mathbf{x}), D(-\mathbf{x}))$. Here $\mathbf{d}(\mathbf{x}), D(-\mathbf{x})$ indicate the motion at the site $\mathbf{x}$ and not including that site respectively. This can be decomposed as follows using Bayes' law.

$$p(O_k^i(\mathbf{x})|\mathbf{d}(\mathbf{x}), D(-\mathbf{x})) \propto p(\mathbf{d}(\mathbf{x})|O_k^i(\mathbf{x})) \times p(O_k^i(\mathbf{x})|O_k(-\mathbf{x})) \qquad (4)$$

The first term on the right is the likelihood that the observed motion vector belongs to the class i, and the second term is the prior probability that the pixel belongs to class i given the current class assignments of the blocks nearby. A Gibbs energy field is used to encode the MRF prior that encourages this smoothness in the label field. Using log-likelihood, and knowing the number of classes, this problem can be posed as an energy minimisation exercise. Without going into more details, the algorithm is as follows for class assignment, for N+1 classes, in which the last class is an outlier class with $Z = N + 1$. Consider that the current block to be labeled has the observed motion $\mathbf{d}$.

1. For each class i , evaluate the vector likelihood energy $E_v^i = [\mathbf{d} - \hat{\mathbf{d}}^i]^T \mathbf{M}^i [\mathbf{d} - \hat{\mathbf{d}}^i]$

2. For each class i , evaluate the class smoothness energy $E_s^i = \sum_{m=1}^{8} \lambda_m ||[O^i - O(m)]||$, where $O(m),\ m = 1 \ldots 8$ refers to the current class labels in the 8-connected neighbourhood of the current block.

3. Calculate an outlier class energy $E^Z = \alpha + E_s^Z$, $\alpha = 10$ acts as a 99% outlier threshold on the likelihood of a vector not being in any of the classes.

4. Calculate the total energy for each class assignment $E^i = E_v^i + E_s^i$.

5. Assign the current motion block to the class with minimum energy among the N+1 energies E^i, E^Z.

This process is iterated over the whole frame in a checkerboard scan, 5 iterations are sufficient. Contiguous regions of Z at the end of these iterations are denoted as separate classes. λ encourages smoothness in the label field and a value of 5 is found to be useful. Note in addition, that the colour of the image covered by a particular motion segment can also be used in the formulation of the likelihood.

Figure 4 shows an apparently simple sequence in which two halves of a coconut are being banged together. The PuM in this sequence is clearly in the motion of the shells and the instant of interest is in the sudden stop at the collision. The motion clusters are shown as coloured blocks and the white lines indicate significant edges in the image between blocks. The colour of each motion segment was modelled as a Gaussian and that was used to augment the likelihood energy in the algorithm above. As can be seen

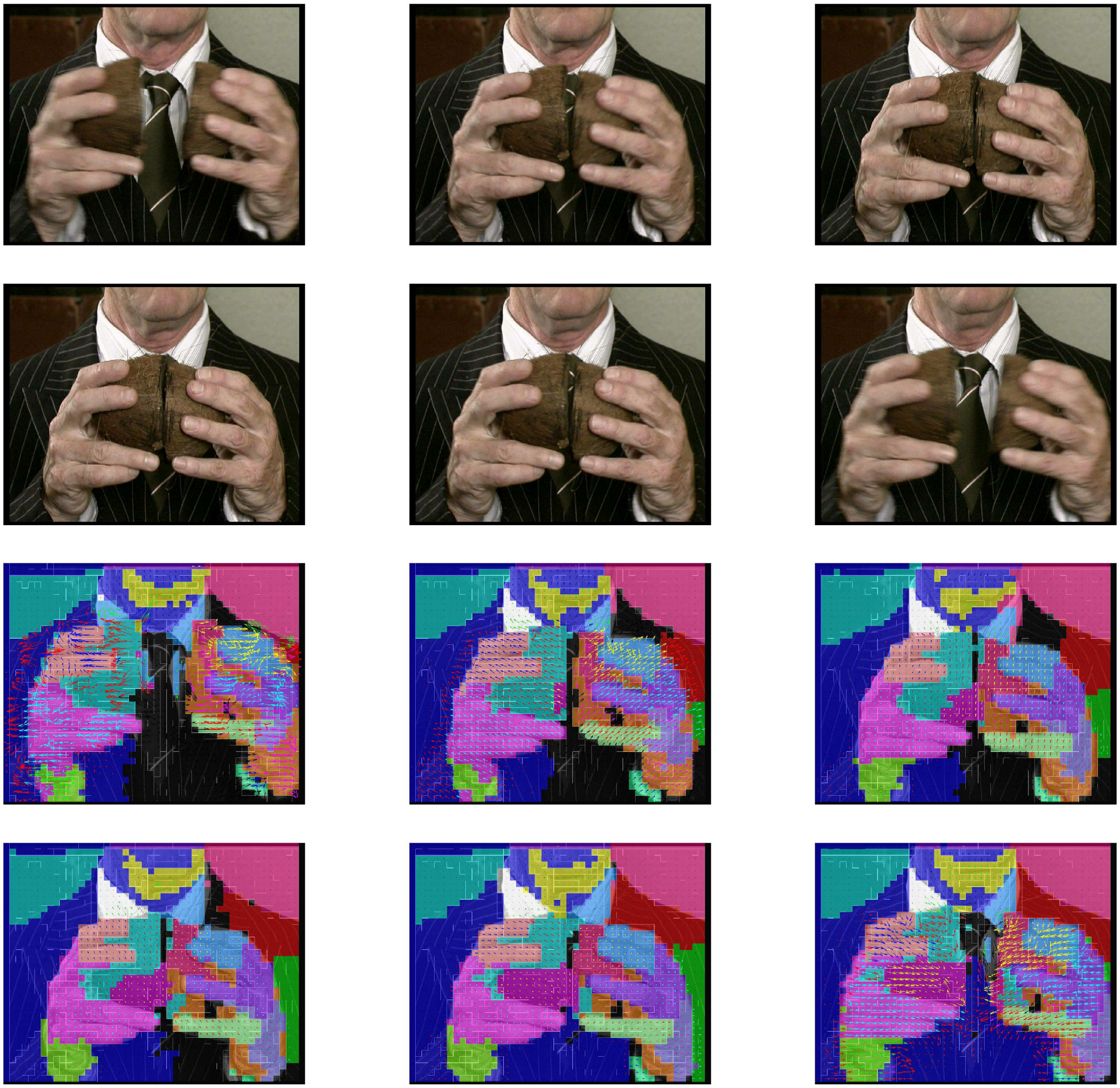

Figure 4: The top two rows show six frames from a sequence showing a coconut shell being beaten. The bottom two rows show the corresponding motion clusters (in colour) and motion vectors estimated using the process described.

the estimated motion of the shells becomes more coherent at the stop itself. The clusters themselves in general maintain their coherency over the sequence. Figure 5 shows a much more complicated example with more complicated semantics. Nevertheless, clusters are forming in a sensible fashion: generally segmenting the body into arms, torso and legs due to motion.

3. MOTION FEATURES

Having delineated objects of interest in each application, motion features can be generated. These features must be extracted with some view to the content being indexed.

3.1 Psy Motion Features

Estimating the amount of rotation in the head can proceed after the centre of rotation is estimated. Consider an image model in the region of the head as follows.

$$I_n(\mathbf{x}) = I_{n-1}(\mathbf{R}\mathbf{x}) + e(\mathbf{x}) \tag{5}$$

where $I(\mathbf{x})$ is the image intensity at site $\mathbf{x}$, $\mathbf{R}$ is the usual rotation transformation $[\cos(\theta),\ -\sin(\theta); \sin(\theta),\ \cos(\theta)]$ and $e(\cdot)$ is a Normally distributed error. Estimation of the rotation θ between the current frame n and the previous frame $n-1$ may proceed with a direct matching technique for the region of interest in the image (i.e. the head), or a parametric fit to the translational motion field previously estimated. Direct image matching is typically more accurate but more computationally intensive. Adopting a parametric fit to the motion field is good enough for parsing.

Given the purely rotational image sequence model above, the displacement $\hat{\mathbf{v}}(\mathbf{x})$ at a site $\mathbf{x} = [i, j]$ can be written as follows.

$$\hat{\mathbf{v}}(\mathbf{x}) = \mathbf{x} - \mathbf{R}\mathbf{x} \tag{6}$$

where the coordinates $\mathbf{x}$ are measured with reference to the estimated centre positions from the previous section. Given observed displacements $\mathbf{v}(\mathbf{x}) = [v_1(\mathbf{x}),\ v_2(\mathbf{x})]$, a least squares estimate for θ is generated by minimising the following error with respect to θ (recalling that $\mathbf{R}$ is a function of θ).

$$E = \sum_{\mathbf{x}} \Big(\mathbf{v}(\mathbf{x}) - [\mathbf{x} - \mathbf{R}\mathbf{x}] \Big)^2 \tag{7}$$

A solution can be generated by differentiating with respect to $\sin(\theta)$ and $\cos(\theta)$ and solving the resulting system of equations. The estimate of rotation $\hat{\theta}$ is then given as

$$\hat{\theta} = \tan^{-1}\left[\frac{\sin(\theta)}{\cos(\theta)}\right] = \frac{\sum_{\mathbf{x}}(iv_2 - jv_1)}{\sum_{\mathbf{x}}(iv_1 + i^2 + jv_2 + j^2)} \tag{8}$$

Figure 6 shows typical results of rotation estimation for one sequence using this method. The images in figure 6 are inserted roughly where they occur on the timeline. They show that the direction of rotation is correctly estimated since the sign of the rotation tracks correspond to the direction of rotation of the head.

3.2 PuM Features

PuM features are derived from the foreground motion map and the local motion clusters. Principal potential editing indicies are located at points in the video that depict a *sudden stop or sudden starts*, i.e. a sudden decrease in the amount of local motion. The foreground motion map is used to generate a global motion trace $M(n)$, defined by $M(n) = \sum_{\mathbf{x}}(l_n^f(\mathbf{x}))$. $M(n)$ is the number of pixels undergoing foreground motion in frame n. Figure 7 shows this trace extracted from a complicated sequence showing a dancer. It contains camera motion as well as changes in scene lighting.

As fas as the local motion clusters are concerned, the statistics of the clusters with time i.e. mean motion magnitide and motion covariance yield useful features. Furthermore the life of the clusters themselves are significant as well as the cluster area. Sharp stops should coincide with cluster decline in motion as well as area.

4. PARSING

Indexing the start and end of the useful Psy experimental data is achieved by detecting the onset of significant rotational movements. This implies detection of both significant rotation and a discontinuity in the head rotation estimate. Figure 6 shows an example of this for one sequence. Significant rotation is detected when the rotation estimate from head tracking is greater than 2.0 standard deviations away from the mean arm angle over the whole active portion of video. Figure 6 shows the detected regions of significance (delineated by the short blue stems) as well as ground truth for these sequences (estimated visually). The start and end of each rotation is estimated to within 5% of the ground truth (generally to within 1 second). These measurements have been made over processing of one hour of material.

4.1 PuM

Detection of Percussive events for editing is decidedly more difficult. The foreground motion trace for the dancer sequence is shown at the top of figure 7. To remove noise in the signal it is processed using Savitky-Golay filtering with a polynomial of order 3. Minima in the smoothed motion trace are identified by consideration of the sign of its first derivative. The $s(n)$ sign signal is given by $s(n) = \text{sign}(\hat{M}(n) - \hat{M}(n-1))$.

Minima in the smoothed motion trace correspond to those points in which $s(n)$ changes from -1 to 0 or 1. To account for residual noise in the signal $s(n)$, it is median filtered with a filter of size 3. The transitions are then detected in the median filtered signal. Transitions where the median filtered $s(n)$ differs in value from the original $s(n)$ are discarded.

At this stage, the signal can be considered as a series of peaks, corresponding to the regions between local minima. Each peak is expected to correspond to a movement in the video, and it is at the end of each peak that the percussive motion is found. Therefore, each peak region is assessed for its ‘peakiness’, by comparing the ratio of the peak ascent to the peak descent. The ascent of a peak is the difference between its maximum value and its starting value, which the descent is the difference between the peak value and the final value. Peaks whose peakiness is less than 20% are discarded, as the end of these peaks probably do not correspond to percussive motion. Figure 7 shows how well this feature alone works for PuM detection in this signal.

The use of the local motion clusters to yield motion magnitude for the coconut sequence is shown in figure 8. Each line is a plot of motion magnitude for a single cluster over its lifetime. As can be seen, the decrease in velocity of clusters corresponds exactly with suitable edit points (shown as dotted lines). In that simple example edit points are located with 100% reliability with this method. The situation with

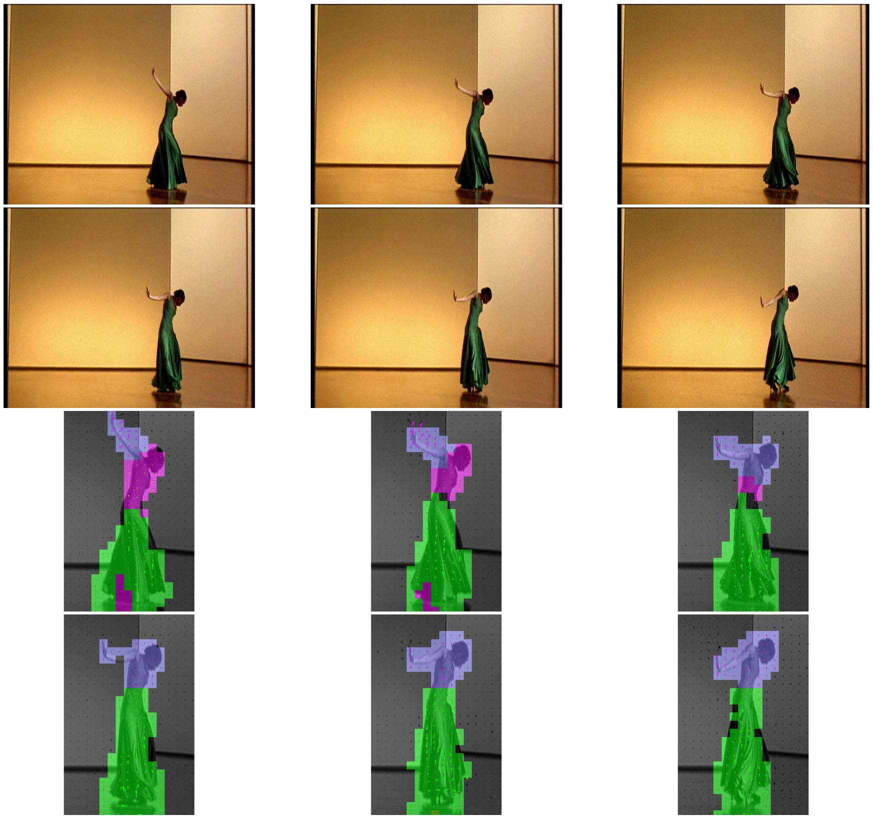

Figure 5: The top two rows show six frames from a sequence showing a dancer. The bottom two rows show the corresponding motion clusters and motion vectors estimated using the process described. A zoom on the foreground region is shown.

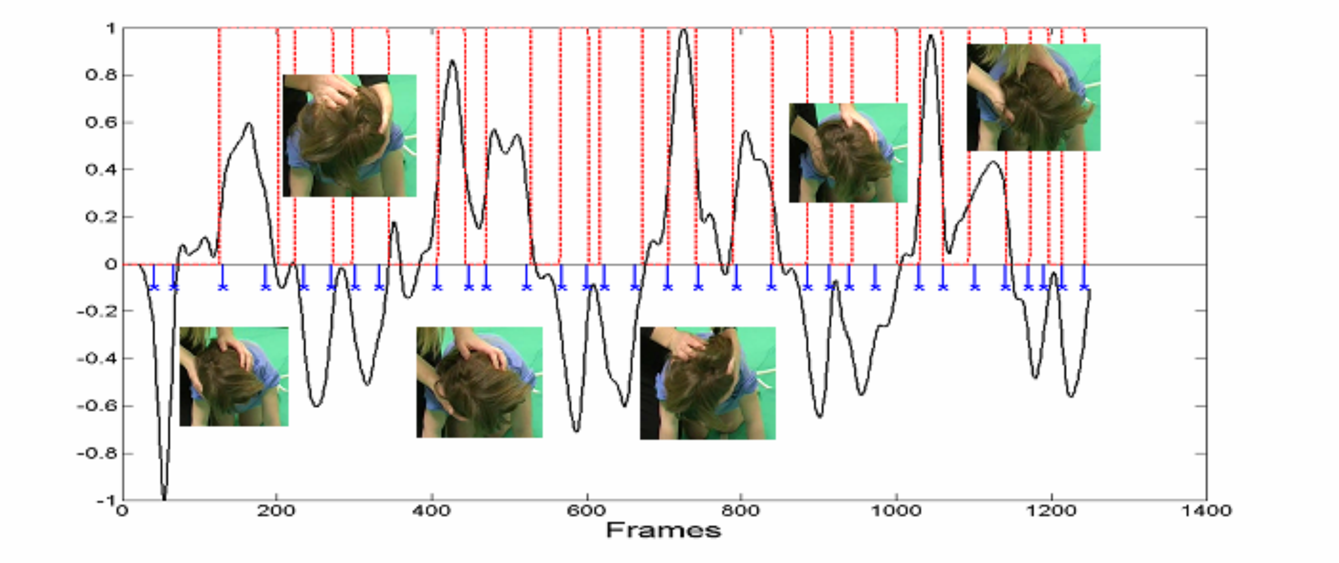

Figure 6: Estimated rotation versus time in frames. The solid line (black) represents a smoothed graph of the inter-frame rotation angle, the dash line (red) represents the manually created ground truth indicating when rotation occurs. The short solid line (blue stem) respresents the estimated start of rotation. The regions detected as containing rotation agree with the ground truth very well. On average less than 5% of frames are missed, and there are less than .01% falsely detected frames.

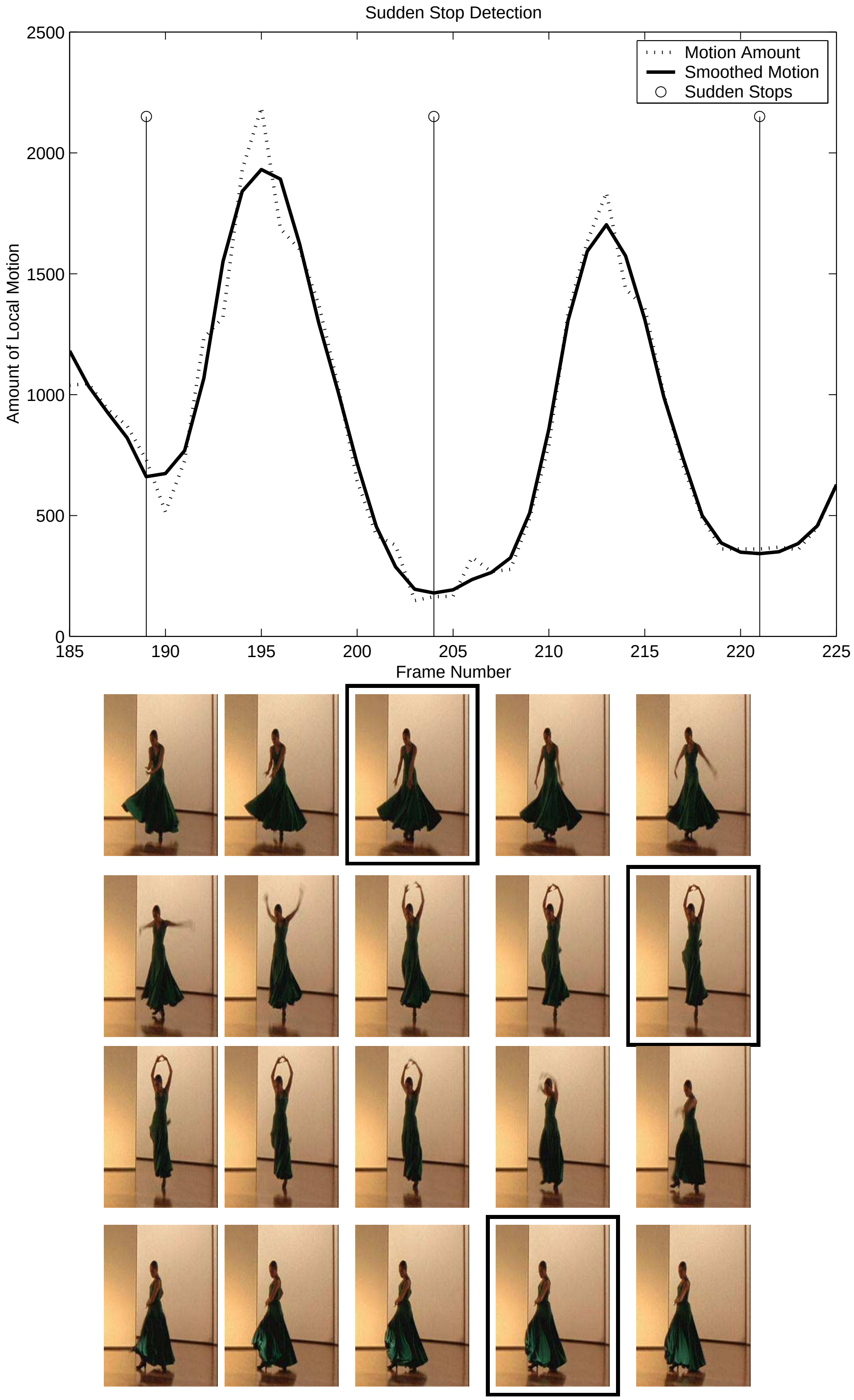

Figure 7: *A motion trace and the corresponding frames from a sequence depicting a dancer. Sudden stop locations are identified by vertical lines traversing the motion trace, and by a black border surrounding the corresponding video frames. It can be seen that sudden stops are identified at locations which constitute a parsing of the phrases in the dance.*

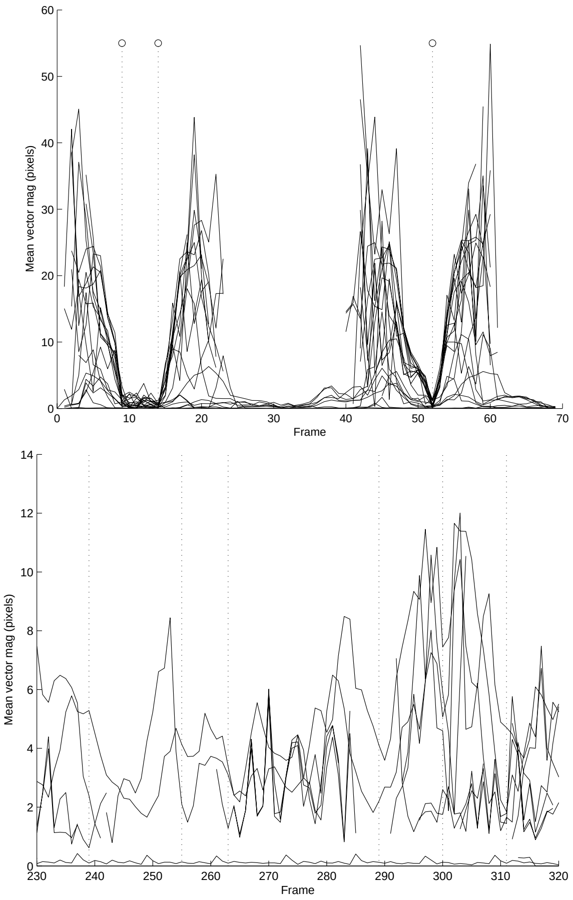

Figure 8: Cluster mean motion magnitude versus frame number for the coconut sequence (top) and the dancer sequence (bottom). Visually assessed edit points (ground truth) are shown at dotted lines.

the dancer is decidely more complex as shown in the bottom of the same figure. The vector magnitude of clusters does indeed decline and show minima at edit points but there are numerous points that attract false alarm. In this, more complicated sequence the local motion clusters can be used to validate the minima discovered in the foreground motion trace.

Over 10 mins of dancing, PuM events are detected with a precision of 70% and recall of 60% in general. This is in comparison with visually assessed edit points which itself is not a reliable ground truth.

4.2 Audio/Video Synchronisation

The validity of the selected edit points is best assessed by use in the application itself. Audio beats are first extracted from the new piece of music using an algorithm first described by Scheirer [16]. The audio signal is split into six frequency bands, and each bandpass signal is fed into a bank of 100 tuned resonators. The resonance frequency showing the strongest response over all six frequency bands is then chosen as corresponding to the tempo of the music.

Given the PuM edit points detected above, the idea is to warp the timeline between visual edit points to coincide with the extracted audio beats from the new piece of music. The number of frames of the output video to be generated to the time of the next beat in the new music track, designated F_{BP} is therefore known. The number of input video frames to the next edit point F_{EP} is also known, having been found using the techniques described above.

Three synchronisation strategies must be adopted depending on the relationship of F_{EP} and F_{BP}.

- $F_{EP} \leq F_{BP}$: In this case, it is desirable to stretch the section of the input video between the two edit points so that it is of exactly the same duration as the time between beat points. This is achieved simply by repeating frames to attain the required frame rate of F_{EP}/F_{BP}. More sophisticared frame interpolation is possible.

- $F_{EP} > F_{BP}$: In this domain it is required to temporally compress the video segment extending to the next edit point so that it fits exactly between the two beat points. This is achieved by discarding frames from the input video. The input video can either be copied at a rate of F_{EP}/F_{BP}, or at a rate that increases exponentially as processing gets closer to the beat point. This second method results in very strong accentuation of the edit point frame.

- $F_{EP} \gg F_{BP}$: Temporal compression of video results in a highly unnatural final effect for high rates of acceleration. To avoid introducing this effect, the next beat point is not accompanied by an edit point frame if it is so far away that acceleration by a factor greater than a threshold (3 used here) would be necessary.

Examples of videos generated using the fully automated application are available at `http://www.mee.tcd.ie/~hdenman/RAVE/`. They show that despite the numerically moderate performance of the PuM detection process, the resulting effect is convincing.

5. DISCUSSION

Ground truth for PuM is difficult to acquire. The notion of PuM itself depends on the perception of the human observer. In addition, semantically relevant objects do not necessarily consist of coherently moving regions. A good example is the dress of a female dancer. The mechanics of cloth imply that those parts close to the body might have a motion different from other parts. Hence the skirt at the waist changes motion with the dancer body while the skirt nearer the floor might still be undergoing smooth motion due to the previous body behaviour. Therefore for the PuM application the implicit motion feature, the foreground motion area, contains more utility than the object based approach. The object based approach can be used to validate edit points estimated with the implicit feature, in a hybrid system. This is because the PuM scenario contains complicated semantics that are not quite well understood.

In the Psy study, the opposite is true. It is a well constrained environment and hence object based motion analysis achieves exactly the desired effect. Parsing and content analysis is therefore facilitated by the strong semantic understanding that can be brought to the application.

The paper has shown that discontinuities in various motion features contain rich information for content analysis.

The remaining issue will always be whether there is sufficient understanding of the semantics of the media for motion features to correspond exactly to a useful parsing.

6. REFERENCES

[1] J. Boreczky and L. A. Rowe. Comparison of video shot boundary detection techniques. In *SPIE Proceedings of Storage and Retrieval for Image and Video Databases*, pages 170–179, 1996.

[2] P. Bouthémy, M. Gelgon, and F. Ganansia. A unified approach to shot change detection and camera motion characterization. *IEEE Transactions on Circuits and Systems for Video Technology*, 9:1030–1044, 1999.

[3] H. Denman, N. Rea, and A. Kokaram. Content-based analysis for video from snooker broadcasts. *Journal of Computer Vision and Image Understanding, Special Issue on Video Retrieval and Summarization*, 92:141–306, November/December 2003.

[4] A. Ekin, A. M. Tekalp, and R. Mehrotra. Automatic soccer video analysis and summarization. *IEEE Transaction on Image Processing*, 12(7):796–807, July 2003.

[5] S. Goddard. *A Teachers Window into a Child's Mind, A non-invasive approach to solving learning and behaviour problems*. Fern Ridge Press, Oregon, 1996.

[6] K. S. Holt. *Child Development: Diagnosis and Assessment*. Butterworth-Heineman Ltd, 1991.

[7] R. S. Illingworth. *The Development of the Infant and Young Child: Normal and Abnormal*. Churchill Livingstone, 8th Edition, London, 1983.

[8] L. Joyeux, E. Doyle, H. Denman, A. C. Crawford, A. Bousseau, A. Kokaram, and R. Fuller. Content based access for a massive database of human observation video. In *Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pages 46–52, 2004.

[9] A. Kokaram and P. Delacourt. A new global estimation algorithm and its application to retrieval in sport events. In *IEEE International Workshop on Multimedia Signal Processing*, October 2001.

[10] A. C. Kokaram. *Motion Picture Restoration: Digital Algorithms for Artefact Suppression in Degraded Motion Picture Film and Video*. Springer Verlag, ISBN 3-540-76040-7, 1998.

[11] M. McPhillips, P. G. Hepper, and G. Mulhern. Effects of replicating primary-reflex movements on specific reading difficulties in children; a randomised double-blind, controlled trial. *Lancet*, 355:537–541, 2000.

[12] J.-M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of Visual Communication and Image Representation*, 6(4), December 1995.

[13] P. Peer, J. Kovac, and F. Solina. Human skin colour clustering for face detection. In *EUROCON 2003 - International Conference on Computer as a Tool*, 2003.

[14] N. Rea, R. Dahyot, and A. Kokaram. Modeling high level structure in sports with motion driven hmms. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 3, pages 621–624, May 2004.

[15] N. Rea, R. Dahyot, and A. Kokaram. Semantic event detection in sports through motion understanding. In *3rd International Conference on Image and Video Retrieval (CIVR 04)*, July 2004.

[16] E. D. Scheirer. Tempo and beat analysis of acoustic musical signals. *Journal of the Acoustic Society of America*, 103(1):588601, 1998.