Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2009, Article ID 924287, 13 pages
doi:10.1155/2009/924287

Review Article

On Parsing Visual Sequences with the Hidden Markov Model

Naomi Harte, Daire Lennon, and Anil Kokaram

School of Engineering, Trinity College Dublin, Dublin 2, Ireland

Correspondence should be addressed to Naomi Harte, nharte@tcd.ie

Received 18 August 2008; Revised 21 January 2009; Accepted 17 June 2009

Recommended by Riccardo Leonardi

Hidden Markov Models have been employed in many vision applications to model and identify events of interest. Their use is common in applications where HMMs are used to classify previously divided segments of video as one of a set of events being modelled. HMMs can also simultaneously segment and classify events within a continuous video, without the need for a separate first step to identify the start and end of the events. This is significantly less common. This paper is an exploration of the development of HMM frameworks for such complete event recognition. A review of how HMMs have been applied to both event classification and recognition is presented. The discussion evolves in parallel with an example of a real application in psychology for illustration. The complete videos depict sessions where candidates perform a number of different exercises under the instruction of a psychologist. The goal is to isolate portions of video containing just one of these exercises. The exercise involves rotating the head of a kneeling subject to the left, back to centre, to the right, to the centre, and repeating a number of times. By designing a HMM system to automatically isolate portions of video containing this exercise, issues such as the strategy of choice of event to be modelled, feature design and selection, as well as training and testing are reviewed. Thus this paper shows how HMMs can be more extensively applied in the domain of event recognition in video.

1. HMMs in Event Recognition

Hidden Markov Models (HMMs) offer a powerful framework for temporal modelling of features extracted from time varying signals. Over 30 years of active research in speech recognition has yielded a core set of tools for feature extraction, training, and recognition, that are well established as the cornerstone of successful speech recognition systems. HMMs have been adopted by the vision community for event recognition in a more cautious manner. Their use gradually moved from augmentation of speech recognition systems with visual information [1], to recognition tasks in video where models are trained on video features alone [2]. This increasing complexity of tasks has echoed the history of speech recognisers in their evolution from isolated word tasks to unconstrained continuous speech recognition.

The use of HMMs in video event recognition takes two approaches: (1) to classify presegmented portions of video (e.g., by shot cut detection), as one of a defined number of classes, (2) to simultaneously, jointly parse and identify events within a continuous video stream. For clarity, this paper will use the term *Event Classification* to describe situations where events of interest are already isolated in time and the task is to identify them as one of a fixed set. Hence, this includes the two pass approach where one algorithm is employed to parse the video and HMMs classify the segments. The term *Event Recognition* will refer to cases where the event is parsed and classified jointly. This important distinction is rarely made in the literature and is central to the theme of the current work. The difference between these two tasks is illustrated by considering the analogy of performing isolated word recognition with HMMs and employing HMMs in continuous speech recognition.

Much of the existing work in HMM-based modelling of visual events involves human motion: a user's hands making a specific gesture in sign language; a cricket bat hit; a goal in a soccer match. As will be discussed in later sections, HMMs have been successfully applied in a small number of event recognition systems, in well defined domains, where human motion is very constrained. Whenever the human motion in the events of interest is more natural and unconstrained, the use of HMMs is typically confined to simply classifying presegmented portions of video. This seeming reluctance to use HMMs for recognition means many potentially suitable applications miss out on the full power of the HMM framework. A greater understanding of the potential of the HMM can extend their application and hence avoid more complex multipass strategies commonly used for event recognition.

Thus, the purpose of this paper is to explore issues involved in building a HMM-based visual event recognition system. The intended contribution is not to further the already extensive theory of HMMs, but rather to take a fresh look at methods already available to visual event recognition and demystify, even encourage, their application. Previous systems using both event classification and event recognition are discussed, highlighting the practical aspects. When embarking on HMM parsing system design in video, it can be difficult to find practical advice. This paper uses an example from a psychology study to give such advice. The style of the paper departs from the classic structure whereby a complete literature review is presented up front, followed by current work. Previous work is instead discussed for each aspect of HMM system design under consideration, immediately followed by how that aspect is considered for the present HMM system. The intention is to make the relevance more immediate for the reader. The aspects of a HMM system considered are: choosing events to model; feature set and HMM topology; and training and evaluation of the system.

A basic familiarity with the use of HMMs and associated terminology is assumed [3]. Detailed work on HMM-based event recognition in the domain of video for psychological assessment of children is presented. This system aims to exploit the successful approach of speech recognition systems in building recognisers. It is considered that the use of HMMs as the mainstay in event recognition in video depends on a number of issues: a feature set that accurately captures the temporal evolution of the event of interest; availability of suitable training data; an appreciation of how and whether the state occupancy corresponds to tangible parts of the event being modelled; and how to constrain the task in terms of an event grammar. Section 2 discusses matching HMMs with events in a new framework and introduces the psychology videos used in the experiments reported in the paper. Section 3 considers the range of visual features used in HMM recognition systems and presents the feature set used to detect rotation events in the current work. Section 4 then focuses on the training and evaluation of HMM recognition systems, including choice of model topology. Results for the psychology video event recognition system are presented in Section 5.

2. Choosing the Event and HMM Framework

The first step in developing a HMM framework is to identify the event to be modelled. This requires careful consideration of all the material that will be encountered in the video sequences. The equivalent in a speech recognition system is identifying whether models are word or phoneme level, and what rules govern how one word/phoneme follows the next. Visual events that evolve in a predictable manner over time and lend themselves to a Markovian model can potentially fully exploit a HMM framework. Unsurprisingly, the earliest use of HMMs for event classification involving human motion in video was in the sports domain [4] because, much like speech, many sports have well established rules and are highly structured. This inherent structure is present at two levels. The first is within a single event of interest, for example, the typical motion of the player during a serve in tennis. It is also seen in the sequences of events, for example, the serve-volley on grass tennis courts. This predictable structure is well modelled by HMMs and the supporting framework for Viterbi recognition. Borrowing from the terminology of speech recognition, a *lexicon* and *grammar* for specific sports such as tennis, basketball, snooker, and cricket can be easily constructed. Ivanov and Bobick [5] refer to this as the primitive components and structure of an activity, respectively.

A wide range of visual sports events have been modelled with HMMs to date. In one of the earliest attempts to classify events involving body motion, Yamoto et al. [4] investigated human action recognition using HMMs to avoid explicit geometric modelling of the human body. HMMs were used to classify tennis events from recorded footage into one of 6 tennis strokes. Petkovic et al. [6] use the same 6 events in their work on tennis footage. The system presented by Kijak et al. [7] takes the classification of events in tennis to a higher temporal level using a hierarchical HMM approach. Four distinct HMMs model the tennis units: missed first serve, rally, replay, and break. The output of the Viterbi recogniser is then used to infer structure at higher levels of point, game, set, and match level. This represents a move towards segmentation at a higher level. Kolonias [8] presents another tennis highlight system which uses a hierarchical analysis of points. A switching HMM approach is used to model first serves, second serves, aces and rallys.

Other sports with events suited to HMM modelling include baseball, soccer, and snooker [9–11]. Chang and Gong [9] used HMMs to classify four types of baseball highlights: nice hits, nice catches, home runs, and plays within the diamond. The system first segments a game video into seven types of scene shots: pitch view, catch overview, catch closeup, running overview, running closeup, audience view, and touchbase closeup. Assfalg [10] attempted to classify three soccer highlight events from video footage: penalties, free kicks, and corners. In Rea [11], HMM shot classification was performed on snooker footage. Four categories of events were classified using HMMs: shot to nothing, break building, conservative play, and snooker escape.

All these systems employ a HMM framework to classify segments of video as belonging to one of a number of possible categories of event. In all cases, the segmentation or isolation of the portion of video under examination is either assumed as given or incorporated as an independent preprocessing stage in the overall system. Such an approach works well in cases where the segmentation stage is inexpensive and robust, for example, in snooker where the camera change is a highly reliable boundary for an event. Unless events are accurately delineated, HMM event classification will remain prone to errors and is not as versatile as fully automated highlight extraction demands in reality. Event *recognition* overcomes this problem.

Event recognition in video using HMMs is not new. The American sign language recognition system presented by Starner [12] fully exploits the HMM framework in a manner most similar to their use in speech recognisers. A 40 word lexicon was used with a gesture corresponding to a word. Each word was modelled with a HMM. HMMs have also been successfully employed in handwriting recognition [13] and lip-reading systems [14]. All these tasks lend themselves well to HMM-based recognition. They have a striking similarity to the speech recognition problem: there is a finite vocabulary, which despite inter- and intra-person variability, is reproducible; how events follow one another is strongly predictable (i.e., the task has a grammar); each event has an evolving temporal structure well modelled by left-to-right HMMs. Such tasks can fully exploit the existing elegant mathematical framework built around HMMs. The path to take in employing HMMs in these problems is clearer—it is not unchartered territory.

The use of HMMs in event recognition where the events involve visual material with less structure is less common. Morguet and Lang [15] present a system for spotting 12 hand gestures in a continuous video stream. The problem is similar to keyword spotting in the audio domain. Boreczky and Wilcox [16] presented a system for video segmentation which concentrated on the task of detecting shots, shot boundaries, and camera movements within shots. By using a HMM with 7 states to model shots, pan and zoom, and transition segments between shots (i.e., cuts, fades, and dissolves), a standard Viterbi algorithm could yield a segmentation on unseen video. Recall of 90–97% was achieved with precision of 79–86%. The choice of event here is interesting as it is the transitions between the events being sought in the video stream that are modelled with the HMM. Cuntoor et al. [17] present a system that jointly segments and classifies events but the number of events in an unseen trajectory needs to be suitably controlled by a scale parameter. The current system, as will be shown, needs no such constraints. Peursum [18] uses the inference of missing data to segment higher level activities into lower level actions. The activities modelled are temporally highly structured. Ivanov and Bobick [5] present a visual event recognition system employing HMMs with recognition at two levels. The first level is based on low level features and then a stochastic context-free grammar is used to parse candidate event sequences by exploiting a priori knowledge of the domain. This work is the most similarly motivated in literature to the current system, the essential difference being that the current work uses context in parallel at the Viterbi recognition stage in a manner most similar to continuous speech recognition systems. This is discussed more fully in Section 3.2.

Robertson [19] recently presented a system for recognition of human behaviour in video systems where HMMs are used at the highest level to model sequences of actions to identify certain behaviours. Thus the HMM inputs and outputs are distributions over action types rather than low level visual features as in the systems considered previously. This raises the question of whether HMMs are best exploited as part of an overall event detection scheme or whether the HMM itself is powerful enough to parse and classify events in a single pass. Indeed, HMMs are used in human activity recognition in [20, 21] as part of overall systems employing Support Vector Machines (SVMs) and multilayer perceptron network layers, respectively.

How the HMM is best used is dependent on how defined the task is, availability of training data, and computational considerations. The HMM framework can be the primary tool for modelling visual events. An initial identification of the structure and rules, if any, of the events allows the definition of the lexicon and grammar for the framework. This will help decide whether HMMs are required for each event, for example, [6] or whether states within a HMM model each event, for example, [16].

2.1. Rotation Events in Psychology Videos. For the example application using video material from a psychology study, the initial step thus was to choose the events to model with HMMs. Some background to the project is necessary at this point. The video material in this work is from a scientific study of the retainment of primary reflexes from infancy in dyslexic children [22]. The hypothesis is that certain reflexes can be triggered and observed in young dyslexic children [23–25]. Specific exercises are performed in order to trigger a particular reflex. One hundred and fifty children were recorded performing fourteen exercises at each of three sessions, with a session taking at least half an hour to complete. The video is then analysed by psychologists. The event recognition system designed in this study is focused on just one of these exercises: the Asymmetrical Tonic Neck Reflex (ATNR) exercise. This involves getting the child on all fours and rotating the head of the child to the left, back to centre, to the right, to the centre, and repeating four times as shown in Figure 1. The aim of this exercise is to look for the primary infant reflex, which in this case is a bend at the elbow of the child during head rotation. The extent of this bend is hypothesised to be proportional to the severity of the dyslexia present. The full outcome of this study in terms of assessing this hypothesis is detailed by Doyle in [26]. The work reported in this paper was concerned with automatically identifying portions of video containing this exercise. The remainder of the video can contain a variety of material such as the child waiting for the exercise to begin, the instructor explaining what will happen, preparing the child for the exercise, or no activity at all.

Given that there is a large amount of video material and that children are not always the most cooperative of subjects, the ability to automatically isolate periods of video containing head rotation in the child would significantly ease the task of later assessing the children in this exercise. This requirement directs the choice of event in this system. The main event of interest is a rotation event. The simplest parsing of the video would thus be as rotation or nonrotation events. This would result in two models (System 1): $\mathcal{R}$ to model rotation; $\overline{\mathcal{R}}$ to model all other events, that is, nonrotation. Another viewpoint suggests that there are actually at least 3 events: child pose setup; pause between head rotations; head rotations. This gives the three model

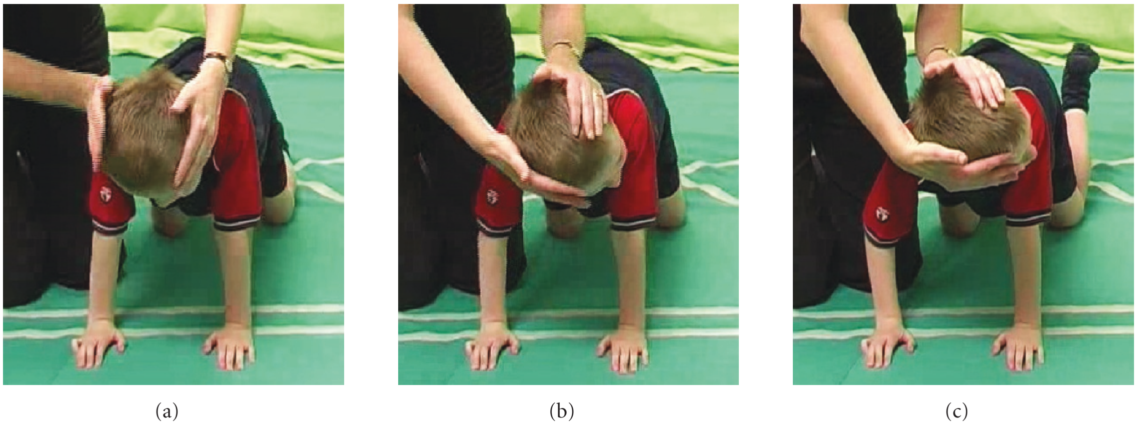

FIGURE 1: The event to be identified. ATNR exercise with head rotation evident. Note the bend in the elbow of the child in the third frame.

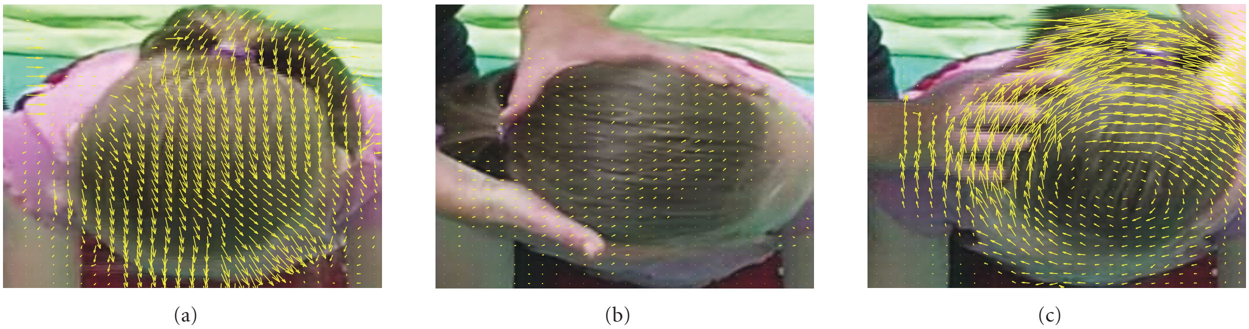

FIGURE 2: Example frames representing the three possible classes of event. Motion vectors have been superimposed. The first image shows random motion of the head of the child up and down. The middle image shows a pause where there is little or no motion. The last image shows a clockwise head rotation.

system (System 2) with HMMs $\mathcal{CPS}$, $\mathcal{P}$, and $\mathcal{R}$. Example frames are shown in Figure 2 where the first frame shows an example of child pose setup, the middle frame shows a pauses between head rotations, and the last frame shows a head rotation.

The order of events in both Systems 1 and 2 is defined by a task grammar. This task grammar is simple and represented diagramatically in Figure 3. The philosophy is similar to that of Ivanov and Bobick's Stochastic Context-Free Grammar [5]. In that system, each HMM models a primitive event. In the recognition phase, each of the HMMs identify the part of the trajectory of a structured event (comprised of the primitives) that they best match. A parser then attempts to find the most likely interpretation of the event set. In the current work, the system exploits the use of a task grammar and Viterbi recognition based on a token passing paradigm [27]. This has the advantage of integrating both the feature evolution and event evolution in a single recognition phase.

It is important to pause here and emphasise again that there is a difference between states in a HMM and the task grammar governing the temporal order of events. This would be familiar to those with a background in HMMs for speech processing but is nevertheless an important distinction. Systems 1 and 2 are alternate grammars constraining different ways of thinking about the whole experiment or the whole temporal evolution of the data itself. Thus system 1 assumes that there are only 2 events following each other throughout the experiment. System 2 assumes that there is an additional event possibly occurring between the two main rotation events. It is these "grammar" models that have the potential to provide the implicit parsing of the data stream into events. The boxes in Figure 3 are "events" not "states" in the sense that we use for the parameterisation of a HMM. The HMM itself is used to model the data stream represented as a particular event. Thus while the $\mathcal{R}$ event is ongoing, the temporal evolution of the feature vectors during the $\mathcal{R}$ event is modelled by HMM-1 say, while the evolution of features during a $\mathcal{CPS}$ event is modelled with HMM-2 say. It is inside HMM-1 and HMM-2 that the specification of "the number of states to use in the HMM" arises. Thus HMM-1 may be a 4 state model say, while HMM-2 might be a 2-state model. It is true that the task grammar is a Markov chain, and one is tempted to say that the grammar is like a super-HMM which then employs other HMMs (in the event boxes, $\mathcal{R}$, $\mathcal{CPS}$, etc.) to model the actual data stream. The grammar is not "hidden" in the strict sense, as the evolution is defined and the events which take place before other events are also defined. Hence "super-HMM" is better denoted as the "task grammar".

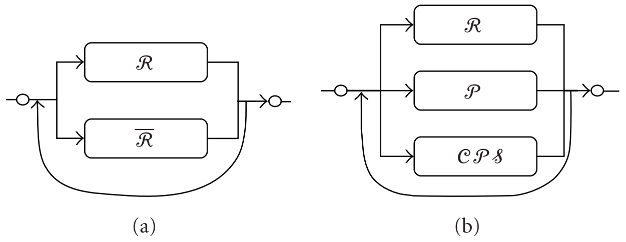

Figure 3: Task grammar for identifying rotation in psychological videos. Diagrams demonstrate the sequencing of events in two possible systems. An event is modelled by a HMM. The diagram on the left shows System 1. The diagram on the right shows System 2.

3. Feature Set and Model Topology

Having identified the event to be modelled by HMMs, the model topology and feature set need to be considered. In some cases the event may have a number of distinct stages, for example, the server body motion in a tennis serve, that can be captured by successive states of the HMM. If the event is always moving forward, a left to right model may be most suitable. In other tasks, the "meaning" of the states may not be tangible and in this case choosing a model with sufficient number of states is important. An ergodic (fully connected) model can be more suited to modelling unstructured events. This has implications for the amount of training data required.

The feature set needs to capture the essence of the event. Cepstrum is a widely employed feature in speech recogntion [28], which captures the spectral trajectory of speech over time and works well over a range of speech recognition tasks. No equivalent to cepstrum has been found for video features. Features are chosen and developed for each application depending on the events being modelled. As noted by Wang [29], visual features typically fall into four categories: colour, texture, shape, and motion. The relative importance of these categories changes according to the event being modelled, for example, the green court in grass tennis, the camera motion in cricket. Features are chosen depending on how reproducible they are over different occurrences of the same event. In speech recognition, whether unseen data will be from different or the same speakers as the training set influences the system, that is, whether the system needs to be speaker independent. Similarly, features must capture this aspect in video event recognition.

In Yamoto et al. [4], a feature vector of size 625 was constructed from mesh features and then vector quantised into one of 72 codewords. The mesh features divide a frame into subareas and measure the ratio of black pixels to the number of pixels in each subarea. Each HMM had 36 states. Petkovic et al. [6] designed a set of 16 features to model the same events, which characterised the shape of the segmented player binary representation. This represented a move towards incorporating visual features to explicitly model shape and motion. The features captured orientation, eccentricity, upper body information, general shape, and sticking out parts. Discrete 8-state HMMs were employed with a codebook of 24 symbols. A subset of these 16 features were identified which yielded the best results where training and test sequences contained data from different players. A 20% improvement in classification over [4] was achieved. This demonstrates how a better choice of feature set and models to cover the events of interest are central to classification success. The features in Kolonias' system [8] are events that are tracked such as the ball bouncing out of court, player position, and shape.

Many systems combine local features derived from a segmented image with global features of the frame. In Kijak et al. [7], shot features are computed for each shot and one keyframe is extracted from the beginning of the shot. The features were shot length, camera motion amount, colour descriptor, and relative player position. The baseball highlight system of Chang and Gong [9] uses a field descriptor, an edge descriptor, grass amount, sand amount, camera motion, and player height. A probabilistic measure is used for the segmentation. The four types of highlights then comprise of defined sequences of scene shots. Each HMM has between three and five states with the transitions controlled by what order of scene shots constitute a particular highlight. Assfalg's soccer highlight system [10] used a discrete 3-state left-to-right HMM model for each highlight type, noting that the three states correspond well to the evolution of the highlights in terms of characteristic content. The features used were a framing term (whether very long shot, long shot, or medium long shot); pan and tilt quantised in 5 and 2 levels. Three extra features to reflect player position were also investigated. In Rea's snooker system [11], the relative position and temporal behaviour of the white ball was considered on the snooker table over the duration of a clip. A colour-based particle filter was employed to robustly track the snooker balls.

Starner [12] used 16 features output from a second moment analysis of segmented hand blobs. Hand occlusions were dealt with by repeating the same features for both hands. A 4-state HMM topology with one skip transition was found to be appropriate. More recent visual recognition systems for sign language [30, 31] incorporate geometric and optical flow features and fully exploit the grammars for the respective sign languages being modelled. The recognition rates for such systems tend to be in the high nineties. Morguet and Lang [15] employed 25-state semicontinuous HMMs to model the hand gestures with features based on Hu moments. Boreczky and Wilcox [16] used a standard histogram distance, an audio distance measure, and a motion estimate. Leahne et al. [32] employ features at the resolution of shots where each shot feature vector contains % speech, % music,% silence, % quiet music, % other audio, % static-camera frames per shot, % nonstatic-camera frames per shot, motion intensity, and shot length. The features were used to classify movie video into dialogues, action sequences, and montages.

3.1. Rotation Features. The feature set for the current application needed to reliably represent head rotation events and distinguish them from other events in the video. The first step was to isolate the child in each frame. Head and arm localisation was performed using skin detection as all the children wore short sleeved tops. The Viterbi algorithm was then used to continuously track the child's arms, once located, to allow the child to be identified within each frame. Full details of this process are available in [33]. The features were local and related to the segmented object of interest (the child), similar to many of the approaches discussed in the preceding section. Intuitively, the features needed to capture the motion of the head of the child, distinguishing rotational movements. A block-based multi-resolution motion estimation scheme [34] was used and the motion vectors for each frame were calculated for each exercise sequence. The blocksize was 9 × 9. Four levels of resolution were used with 10 iterations, a displaced frame difference threshold of 1.0, and 5 iterations of smoothing.

3.1.1. Rotation Centre Stability. The first feature was chosen to capture the stability of the head. All perpendiculars to the rotational motion vectors will intersect the centre of the rotating head. Nonrotational motion vectors should rarely cross the centre of rotation. Plotting the perpendicular lines to the motion vectors in an accumulator array allows an approximate centre of rotation to be found. This is similar to the straight line analysis of Wong et al. for fast rotation centre identification [35]. Using only motion vectors within ±15 pixels to the left and right of the child's arms reduces the vectors to those relevant to the child. During rotational events, accumulator array maxima were stable. They were found to be extremely unstable during nonrotational events. Measuring the euclidean distance for accumulator maxima for consecutive frames, $\mathcal{A}_{\text{dist}}$, was found to broadly give low values during rotation and higher values during nonrotation. This can be seen in Figure 6d showing the evolution of this feature for a sample video. Errors occurred in this observation during low motion events where there were few, if any, contributory motion vectors, causing the centre of rotation to be stable in the absence of rotation. Hence the accumulator array maxima monitoring was insufficient on its own to identify rotational events but provides useful cues.

3.1.2. Curl Related Features. When observing a rotational motion field, it can be seen that in a row of motion vectors, no two motion vectors have the same x and y components. This is true for all rotation and the rate of change in these component values is relatively constant. The curl property C of a vector field is defined in (1). It is a combination of the rate of change of the y velocity components in the x direction with rate of change of the x velocity components in y direction:

$$\mathcal{C}(v_1, v_2, \mathbf{x}) = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \dfrac{d}{dx} & \dfrac{d}{dy} & 0 \\ v_1 & v_2 & 0 \end{vmatrix}, \qquad (1)$$

$$\mathcal{C}(v_1, v_2, \mathbf{x}) = \vec{k}\left(\frac{d(v_2)}{dx} - \frac{d(v_1)}{dy}\right).$$

However, d/dx and d/dy are horizontal and vertical gradients respectively. Also, $\vec{i}$, $\vec{j}$ and $\vec{k}$ are the orthonormal basis for the vector space. v_1 and v_2 are the estimated translational motion vector components in the x and y directions at pixel $\mathbf{x}$. When only one object is rotating in the scene, the curl field will contain a peak located on the centre of rotation, as can be seen in Figure 4 where the dominant peak clearly shows the presence of head rotation. Three extra features were derived from the curl field.

In order to track when rotation occurs, the position and value of the curl field maximum are monitored. The maximum value, $\mathcal{C}_{\max}$, was found to rise and fall consistently during head rotation and vary randomly during nonrotational events (see Figure 6a). It was also noted that the area of the maximum peak, $\mathcal{C}_{\text{area}}$, increased and decreased consistently during rotation (see Figure 6b). Similar to the distance measure on the accumulator array, it was observed that the curl surface maximum position from frame to frame, $\mathcal{C}_{\text{dist}}$, was stable during rotation (see Figure 6c). The curl surface was segmented to find the area of the main peak. Segmentation was done using a watershed algorithm [36] to identify the dominant peak in the surface. To ensure that the segmentation is performed correctly, the absolute curl surface is negated. This ensures that the peak corresponding to head rotation is always the dominant peak on the surface. An example is shown in Figure 5. Curl surface peak tracking alone is not powerful enough to consistently indicate rotation. The feature robustness is affected by the fact that the head of the child is not perfectly circular, the area of rotation can be irregular due to occlusions from the hands of demonstrator guiding rotation, and time varying.

The first and second temporal derivatives of the curl maxima values ($\mathcal{C}'_{\max}\quad \mathcal{C}''_{\max}$) and the curl peak areas ($\mathcal{C}'_{\text{area}}\quad \mathcal{C}''_{\text{area}}$) were used to augment the feature vector. This is a method frequently used in speech recognition where feature trajectories can contain extra information in the feature vector. Derivatives were calculated over a window of ±2 frames. The rate of change of these values during rotation and nonrotation events was observed to be distinctly different. The complete set of features is summarised in Table 1.

The question may arise of why not use Gaussian Mixture Models (GMMs) rather than employing a HMM framework. Bashir et al. present a discussion of this very question in [37]. In the current work, it is the ability of the HMM framework to model both the temporal evolution of the feature set and the inherent uncertainty in the unfolding of rotational events that makes them particularly suitable. As discussed, none of the features alone are capable of reliably and accurately parsing rotation events, though each can be seen to exhibit largely predictable behaviour during rotation and nonrotation. The core emphasis of this paper is the ability to both segment and classify events using a single pass approach. Only the HMM framework offers that potential.

3.2. Model Choice. As explained in Section 2.1 two systems, System 1 (two HMM approach) and System 2 (3 HMM approach), were developed. The number of states per HMM

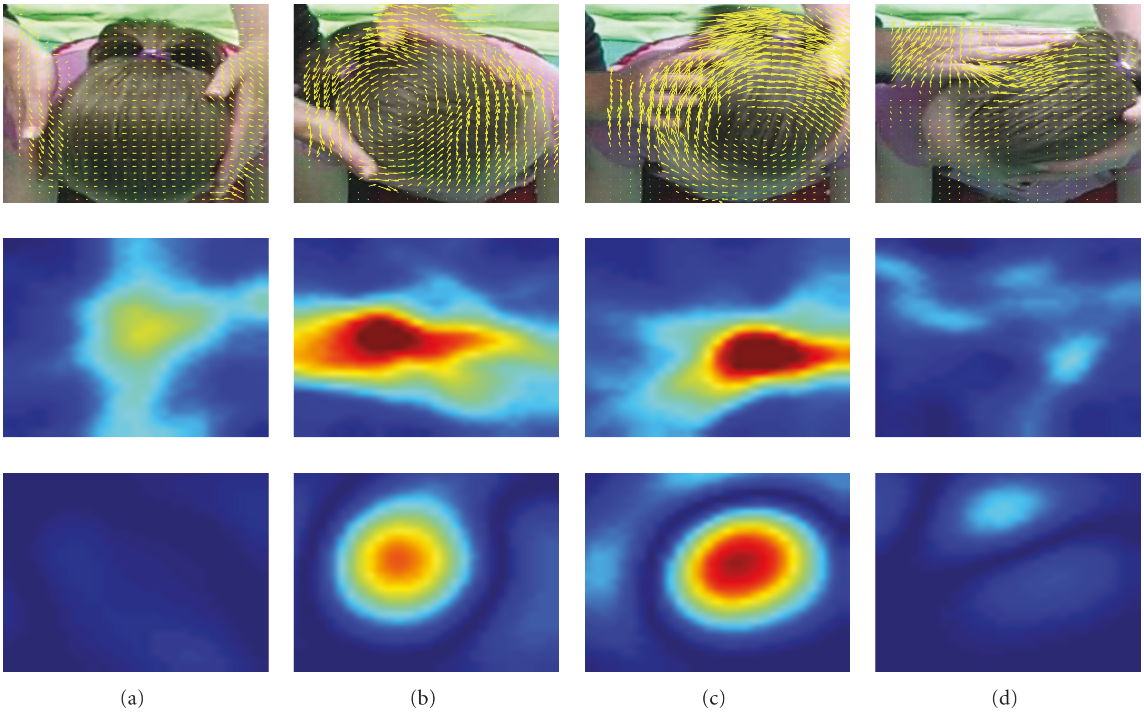

Figure 4: The top four images show 4 frames from a sequence of head rotation. Note the motion vectors. The central four images show the accumulator array for the same frames and the bottom four images show the curl field. All of the above images have been zoomed in to improve clarity.

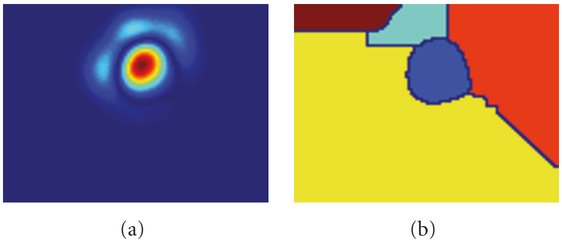

Figure 5: An example curl surface is shown on the left above along with the corresponding watershed segmentation boundaries on the right. A clear delineation is visible between the central peak and the rest of the curl surface.

and the model topology in terms of allowable transitions needed to be chosen. In any system, restricting the state transitions for a given amount of training data increases the effective amount of data available for estimating individual state transition probabilities. The data for the nonrotation event models $\mathcal{CPS}$, $\mathcal{P}$, and $\overline{\mathcal{R}}$ is inherently unstructured, and it is difficult to identify a temporal evolution of features that would be suitably modelled by a left-to-right HMM. Hence an ergodic HMM was always used for these events.

The rotation events should display more temporal structure as the child's head is turned to each side in turn for approximately 5 seconds. Analysis of the videos showed that the head was not always turned in the same direction first however. Hence a strict left, to, right model is not suitable for the rotation events. Despite this, intuitively the rotation event could be thought of as comprising of three elements: head moving clockwise, the head moving anticlockwise; or pauses. A partially connected model was constructed to exploit this temporal event composition. The partially connected model has $3N$ states, where N was 2, 3, or 4. Thus these models had 6, 9, or 12 states. Self transitions and left, to, right transitions were allowed for all states but skip transitions were reduced. The allowable skips were from state N to state $2N + 1$, state $2N$ to state 1 and from state $3N$ to state 1 and $N + 1$. Entry states were reduced to state 1, $N+1$, and $2N+1$ and exit states reduced to states N, $2N$, and $3N$. The motivation for this was to encourage the three groups of states to loosely correspond

Table 1: Feature set.

Feature	Description
$\mathcal{C}_{\text{max}}$	Curl Surface Maxima
$\mathcal{C}_{\text{area}}$	Curl Surface Max Peak Area
$\mathcal{C}_{\text{dist}}$	Curl Maxima Distance from Frame to Frame
$\mathcal{A}_{\text{dist}}$	Accumulator Maxima Distance from Frame to Frame
$\mathcal{C}'_{\text{area}}$	First Derivative of Graph of Curl Surface Max Peak Area
$\mathcal{C}'_{\text{max}}$	First Derivative of Graph of Curl Surface Maxima
$\mathcal{C}''_{\text{area}}$	Second Derivative of Graph of Curl Surface Max Peak Area
$\mathcal{C}''_{\text{max}}$	Second Derivative of Graph of Curl Surface Maxima

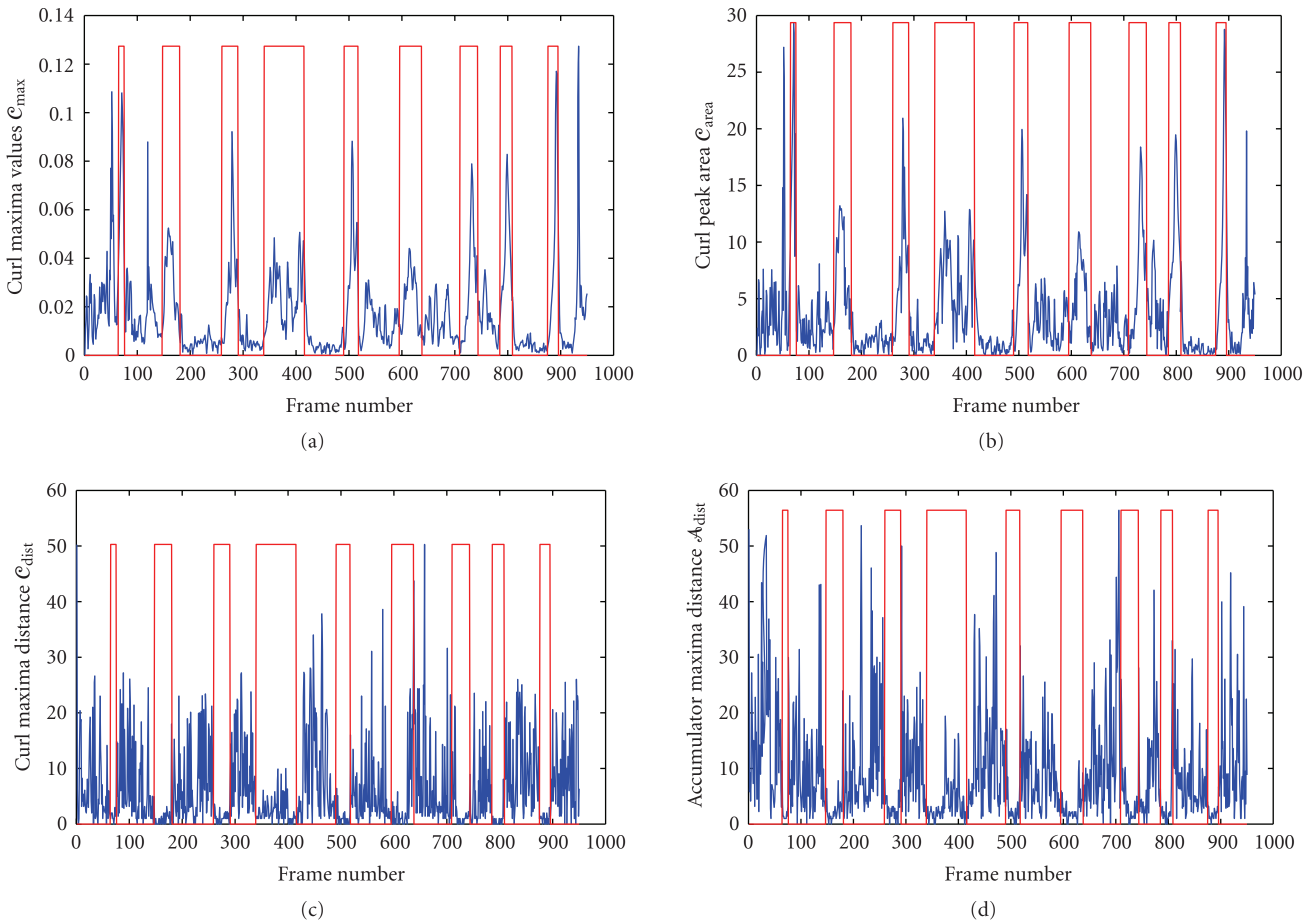

Figure 6: Sample feature evolution for video. Note manual segmentation superimposed where the high level implies head rotation is occuring.

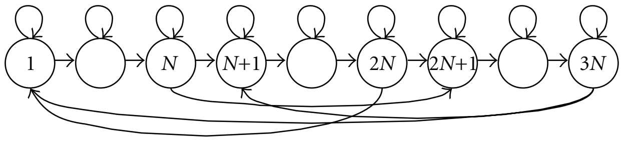

Figure 7: The partially connected model. The example shown has N equal 3.

to clockwise rotation, anticlockwise rotation, and pauses but not to put a strict temporal order on these three constituent elements of a rotation event. Figure 7 demonstrates the $N = 3$ case. Both the use of this partially connected model and a fully connected model were investigated for the rotation events. The fully connected models had between two and twelve states per HMM for all models.

4. Training and Evaluating the System

Having chosen a feature set and HMM topology to model the events of interest, a significant effort is required to train a HMM system. The task of acquiring sufficient labelled training data is one of the factors that can put users off HMMs. To develop a system, the available data set should be divided into training and reference testing data. Generally, a minimum of a 2 : 1 train : test ratio is advisable. The training set needs enough examples of each event to adequately train the number of gaussian mixtures the HMM topology has. The examples must reflect the range of typical occurrences of these events that the system is expected to subsequently correctly identify. The test data should not overlap the training data and also needs to contain sufficient occurrences of all events, such that measured classification or recognition rates are statistically significant. Continuous density HMMs will require more training data than discrete models. Ergodic models will require more training data than left-to-right topologies to adequately train all transitions.

It is important to be aware of the symptoms of insufficient training. Indications of poorly trained models include poor recognition performance, models not converging during Baum-Welch reestimation, and model parameters not moving significantly from initial values (typically from a flat start). A poor choice of features can cause similar problems. Hence a proper systematic evaluation of a new feature set is essential. Training with full covariance models can uncover problems hidden by the common assumption of diagonal variances for the observation densities within states. Over training must also be avoided. Thus in development, it is best to check classification/recognition performance on both training and test material to assure even performance.

It is useful to examine other systems to assess the amount of training data employed. Yamoto et al. [4] used data from three people performing each of the 6 tennis actions 10 times. Smaller data sets were typical in earlier systems as computational requirements were limiting. In Assfalg's soccer highlight system [10], both training and test data were very limited—only 10 shots were used in training for each highlight class and then 10 shots of each highlight type chosen for testing. This task was very small and it is unclear whether the high classification rates would be achievable for a larger-scale experiment. Chang and Gong [9] used 18 hours of footage in developing their baseball system, suggesting that the training was adequate. Classification rates varied from 40 to 71% for the highlights. It could be argued that this system did not fully exploit the temporal modelling capabilities of the HMM framework. If a similar approach was taken in speech recognition, this would be equivalent to performing phoneme level recognition first and then using that phonetic segmentation, inherently prone to errors, to try to infer word level recognition. A HMM approach which jointly modelled both the scene shots and the highlight level of segmentation is possible and would have yielded better performance. This demonstrates that it is not safe to assume that performance problems should immediately be attributed to the quantity of training data available.

The first-person-view sign language system of Starner [12] used 400 sentences to train 4 state HMMs for each gesture. 100 different sentences were used for recognition and over 99% recognition accuracy was achieved. Sentences took the form pronoun-verb-noun-adjective-pronoun (same one), and Viterbi exploited this known grammar structure. With six pronouns, nine verbs, twenty nouns, and five adjective, this is effectively less test data than might first be apparent. With greater computational power now available a decade later, testing on larger datasets is feasible.

Alternatives to conventional HMM training approaches may be worth investigating. Brand [2] considers that the weakness in using HMM for visual event recognition lies in an uninformed choice of models and topology. By minimizing the entropy of state distributions rather than using the traditional Baum-Welch approach to training or individual models for each activity, the internal state machine of the HMM can organise observed activity into highly interpretable hidden states. These states in turn capture the dynamical regularities of the training set. Here event segmentation and classification become a single inference problem and the Viterbi stage alignment automatically yields the event sequence for continuous video streams. This is a very useful approach to the training question when a choice of distinct models for events is unclear. This highlights the issue of how important it is to have a good understanding of the event being modelled and what the HMM framework can offer when modelling image sequences. For instance, Liu et al. further discuss the issue of training in [38] for gesture recognition. Two hand gestures, depiction of a triangle and depiction of a square, are chosen to aid the experiments. They discuss the relative merits of different methods of initialising models at the outset of Baum-Welch training. The triangle gesture has three distinct stages of approximately equal duration, the square gesture four. Hence a 3-stage left-to-right model (including same state transitions) is used to model the triangle and a 4-state left-to-right model used for the square. They compare the transition matrices and observation densities when the data is evenly divided between states and all values hand computed, to the output of Baum-Welch training. That the values are similar, with such constrained data, should not be surprising. If the durations of the gestures were allowed vary, the Baum-Welch algorithm would certainly yield superior results as it is the ability of HMMs to model temporal variability that is their very strength. Initialising Baum-Welch training with good initial estimates should indeed speed up the convergence of the training. With such a small training set (20 samples of a gesture), the fact that random initialisation is not as good as informed initialisation suggests the models are not converging. It is also no surprise that with a highly temporally-structured gesture and a small training set, and that left-to-right models will outperform fully ergodic models.

4.1. Psychology Video System. Twenty three videos with this rotation exercise were available for experiments totalling approximately 20 minutes of footage. High-quality motion vectors were essential, as all features are derived from them. Hence the child had to be sufficiently large in the scene. An average arm separation of approximately 200 pixels was found to be suitable to ensure this. All 23 videos had rotational events manually labelled to supply ground truth data. There was a total of 29429 frames of footage, of which 10046 depicted 107 rotation events. There were 121 examples of nonrotation. Sixteen videos were randomly selected for training purposes, and seven selected for testing. From the outset, there was an awareness that this amount of data might not be enough to adequately train a system and this was kept in mind in assessing results. A full examination of the feature set performance was carried out.

As explained in Section 2.1 two systems, System 1 (two HMM approach) and System 2 (3 HMM approach) were developed. The nonrotational data used in training model $\overline{\mathcal{R}}$ was further subdivided for the two HMMs $\mathcal{CPS}$ and $\mathcal{P}$. The data was segmented on the basis of duration of the nonrotation events: any event lasting longer than 100 frames was classified as $\mathcal{CPS}$. However, $\mathcal{CPS}$ thus included getting the child setup after some restlessness or repositioning the child during the exercise.

Both training and recognition were performed with the Cambridge Hidden Markov Model Toolkit (HTK) [39]. HTK was developed originally for speech recognition applications but can easily be integrated into visual event recognition systems. Standard Baum-Welch training was performed using labelled training data. Interestingly, due to the use of the fully ergodic model, it was possible to take all segments for rotation in a video and join them together. This meant that each video was divided into two subvideos of rotation and nonrotation. The segments containing rotation tended to be significantly shorter than the nonrotational events. The longer training sequences allow a larger number of state path alignments in training. In fact, this method was found to outperform the use of individual segments in this case since the training data set was limited. When using the individual segments, many were quite short in duration and did not allow as full a training of the state transitions. Continuous density models were used as the feature set is inherently continuous in nature.

Recognition was performed on the test data using the standard Viterbi algorithm. The comparison between the HMM and manual segmentations were evaluated using precision and recall rates defined in what follows. A tolerance of 14 frames, roughly half a second was allowed between the HMM output and manual segmentations. This was to allow for human error in noting rotation events, as a human observer can frequently interpret prerotation head translation as rotation. The precision and recall figures as detailed below were calculated at a frame level. The performance figures are calculated in terms of correctly identified frames or true positives (t_{pos}), false positives (f_{pos}), and false negatives (f_{neg}):

$$\text{Recall} = \frac{t_{\text{pos}}}{t_{\text{pos}} + f_{\text{neg}}}, \qquad \text{Precision} = \frac{t_{\text{pos}}}{t_{\text{pos}} + f_{\text{pos}}}. \qquad (2)$$

Precision and recall can be combined into a single measure as the F_1 value:

$$F_1 = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}. \qquad (3)$$

The Viterbi algorithm employed is based on the token passing paradigm of Young [27]. The grammar of the system allows syntactic constraints to be applied to identify only allowable paths. For an unseen video, possible event boundaries are recorded in a linked list structure as the token propagation is performed during the Viterbi alignment. At the end of the video, the path identifier held by the token with the minimum alignment cost allows the best event sequence to be traced back and the corresponding boundaries recovered.

5. Results

5.1. Feature Set Evaluation. As discussed in Section 3.1, there were four basic features and four associated derivative values in the full feature set. A study of the performance of all combinations of $\mathcal{C}_{\text{max}}$, $\mathcal{C}_{\text{area}}$, $\mathcal{C}_{\text{dist}}$, and $\mathcal{A}_{\text{dist}}$ was undertaken. Table 2 shows the subset of these combinations that test set performance is reported for in this paper. Initial tests were performed using two ergodic HMMs, $\mathcal{R}$ and $\overline{\mathcal{R}}$, trained with between two and twelve states with a single mixture per state. The performance on the test set was in line with recognition rates for the training set in all cases. Table 3 details some test results of note for these sets. Figure 9 shows more complete results for these sets where the number of states was varied between 2 and 12. A second-order polynomial line fit was used to show trends for each data set. Note that only data points reported in Table 3 are plotted for sets A, B, and C as performance plateaued at the optimal number of states and is outside the scale of this ROC (Region of Convergence) otherwise. These feature sets only used one feature. This ROC graph gives a good pictorial representation of the recall-precision tradeoff. The results showed that $\mathcal{C}_{\text{dist}}$ and $\mathcal{A}_{\text{dist}}$ have the greatest discriminative ability when combined with either $\mathcal{C}_{\text{max}}$ or $\mathcal{C}_{\text{area}}$. It intuitively makes sense that $\mathcal{C}_{\text{dist}}$ and $\mathcal{A}_{\text{dist}}$ are good features as they relate to the stability of the rotating centre. Both $\mathcal{C}_{\text{max}}$ and $\mathcal{C}_{\text{area}}$ are more prone to error as discussed in Section 3.1.2. The use of second-order derivatives did not improve performance, though the first-order derivative allowed for better recall rates at the cost of precision. Examination of these features revealed that they were quite noisy. An increase in the window over which the derivatives were calculated or suitable smoothing of the feature could improve their contribution. Between 4 and 6 states were found to be an optimal number of states in each HMM. Increasing the number of states further adversely affected precision, even though recall could be improved. Figure 9 clearly shows the steep drop off in precision when the HMM has too great a number of states for the available training data.

Figure 8 shows a manual segmentation compared to a sample output from the HMM segmentation for a short section of video. As can be seen in this example, all periods of rotation were identified but the precision of the start and end points of the events was inaccurate. This result was obtained using feature set H. For this application, precision is less important than recall, as false positives are less important that missed events. The framework is intended to avoid the psychologists having to look at all the video material to search for rotation events. In this case, it would be acceptable to have a solution which had a high hit rate which located the event in time and the psychologist could then accurately identify the start and end of the event for their evaluation. The reported precision and recall rates in Table 3 have an accuracy window of 1 second. If this is increased to a window of 2 seconds, both the recall and precision rates approach 94–95% for the feature sets H, I, J, and K in Table 3.

5.2. Amount of Training and Test Data. As mentioned in Section 4.1, there was an initial concern that the data might not be sufficient to train a HMM framework. This framework is particular insofar as it is a two model system. A larger number of models will result in a more complex system requiring more training data. The rotation event HMM had less training data than the nonrotation event HMM as more of the data represented nonrotation periods. In this way, the event with greater variance received more data as would be desired. Careful note was taken of model convergence during Baum-Welch re-estimation. The recognition performance on the training data and test data was consistent both in precision and recall. Lack of training data will typically cause the performance of a recognition system to collapse when the number of states exceeds the level at which all states are adequately trained.

It is useful to compare the amount of data used to other HMM systems. The TIMIT database is accepted in the speech community as a standard medium vocabulary database used in HMM-based speech recognition systems.

Table 2: Subset of Feature Sets from full testing. ✓ denotes individual feature included in feature vector for a set. Blank entry implies the feature was excluded from that set.

Set	$\mathcal{C}_{\max}$	$\mathcal{C}_{\text{area}}$	$\mathcal{C}_{\text{dist}}$	$\mathcal{A}_{\text{dist}}$	$\mathcal{C}'_{\max}$	$\mathcal{C}'_{\text{area}}$	$\mathcal{C}''_{\text{area}}$	$\mathcal{C}''_{\max}$
A				✓				
B			✓					
C			✓	✓				
D	✓		✓	✓				
E		✓	✓	✓				
F	✓	✓	✓	✓				
G	✓		✓	✓	✓			
H	✓		✓	✓	✓			✓
I			✓	✓		✓		
J			✓	✓		✓	✓	

Table 3: Feature Set Performance for 2-model ergodic HMM framework.

Set	States	Recall	Precision	F_1
A	4	72.2	84.0	77.6
B	4	79.9	82.7	81.2
C	12	90.1	86.4	88.2
D	6	95.8	82.8	88.8
E	7	93.1	83.4	87.98
F	8	81.5	81.7	81.6
G	4	75.1	92.5	82.9
G	6	97.2	77.3	86.1
H	4	76.4	90.8	83.0
H	6	97.5	73.1	83.5
I	7	94.5	76.2	84.3
J	4	83.4	89.8	86.5

Table 4: Feature Set Performance for 3-model ergodic HMM framework.

Set	States	Recall	Precision	F_1
G	7	99.1	81.4	87.7
H	7	96.0	80.1	87.3
I	8	94.7	81.7	87.7
J	9	94.2	79.3	86.1

The common database allows for comparison of results across research efforts. In a HMM speech recognition system using monophone models, 3 states per model with up to 20 mixtures per state for each of 39 phoneme models is typical. Using the specified data in TIMIT, this equates to an average of 600 frames of training data (assuming a 10-millisecond frame rate for feature extraction) available per Gaussian mixture. This system would use feature vectors of 36 features. The training data is 73% of the data, the rest is used for test.

In the psychology videos in this paper, the best results will be seen to be for 6–9 states per model with one mixture per state. The maximum number of features in a vector was 8. Thus there was 776 frames per Gaussian mixture for training the rotational HMM, and 1498 frames for training each nonrotational HMM Gaussian mixture. The most states used was 12, giving 582 and 1123 frames per mixture. The training data is 69% of the total data, the rest is used for test. There are more transitions in the models for the psychology material since it is not a simple left-right model, but the extra data will address that. Thus the data used in both training and testing this system compares well.

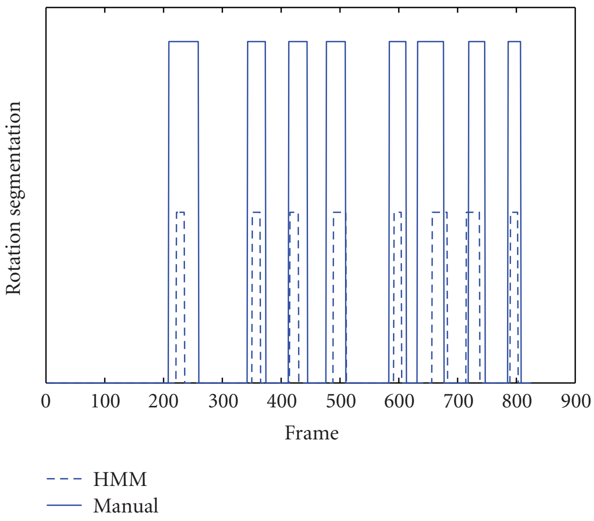

Figure 8: Manual Segmentation (solid line) compared to HMM output (dashed line) for sample video where nonzero values indicate rotation.

To compare with other work in video event recognition, in the previously discussed work of Boreckzy and Wilcox [16], only 6 minutes of training data was used to train 8 Gaussian mixtures (of the 7 HMM states, one had 2 mixtures and all others had a single mixture). At a frame rate of 30fps, this equates to an average of 1350 frames to train a Gaussian mixture. Each test set was 30 minutes of data but the authors were trying to classify 7 different events, whereas the system reported here uses 7 minutes to demonstrate the recognition rate of 2 and 3 events.

It is important to say that the robustness of the system would be improved with a greater amount of training data, especially in the case where the rotation events became less consistent. More exhaustive testing with a larger dataset would improve confidence in the performance of the system, but this initial performance is very encouraging and demonstrates how features and models can be designed to work well in a new event recognition system which is the central tenet of this paper.

5.3. 3-*Model System.*

The results shown in Table 3 refer to the two model ergodic HMM system. As discussed in Section 2.1, the framework could also be posed as a three model system with models corresponding to child pose setup $\mathcal{CPS}$; pauses between head rotations $\mathcal{P}$; head rotations $\mathcal{R}$. This approach was tested using feature set G, H, I, and J as defined in Table 2. Table 4 shows the results for the 3-model system. This system yielded small improvements in both recall and precision rates.

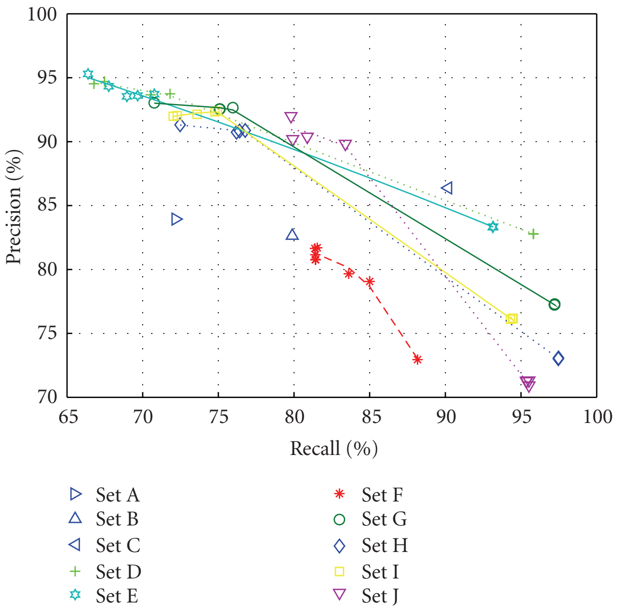

Figure 9: ROC curve showing recall and precision for Set A to Set J inclusive, using between 2 and 12 states. Note that only one data point is shown for set A, B, and C.

5.4. Model Choice. The results presented have used an ergodic model for rotation and nonrotation. The feature sets G, H, I, and J were tested with the partially connected HMM for $\mathcal{R}$ as described in Section 3.2. An ergodic model was still used for non rotation events. The results were not as good as using the ergodic model. F_1 measures of 78.4% and 79.7% were achieved with the 6 state model for feature sets H and J. Checking the Viterbi recognition results from the training data revealed a mismatch in performance for the test data and training data, which in turn prompted a reevaluation of the video material. An insufficient portion of the videos displayed this definitive split of head moving clockwise, head moving anticlockwise, and pauses within a rotation event. Thus an ergodic model was a better choice for this application. This allowed the Baum-Welch training full freedom in exploring the optimal paths within the HMM for modelling the rotation event.

6. Lessons Learned

HMMs have been successfully deployed in video applications for both event classification and event recognition in the past. This paper has used the development of a HMM framework for event recognition in psychology videos to support an exploration of the use of HMM frameworks. The presented system can achieve recall and precision rates of up to 95% for the recognition of the described head rotation events. When considering the use of HMMs for such applications, it can be difficult to find practical guidance on how to approach the task despite the HMM theory being well documented. The first step is to consider the events to be recognised and whether a task grammar is identifiable. This will help identify whether HMMs model events directly or whether it is more appropriate that the states within the HMM will model the events. An appropriate choice of model topology, whether left-to-right or fully connected models are used, should also be a decision made with some reasonable basis. Features can encompass both local and global frame characteristics depending on the nature of the event being modelled. Training data must be representative of the range of manifestations of the events. Modelling a large number of events with significant variation within events requires significantly more training data than the application presented in this paper. When using ergodic models for less structured events, training samples may be joined to increase precision. It is hoped that a greater understanding of such issues will enable the use of HMM-based systems to recognise events in less constrained tasks in video without the need to presegment the video using another method.

Acknowledgments

This work was supported in part by the Science Foundation Ireland under Grant /06/RFP/ENE004, and the Irish Research Council for Science, Engineering and Technology (IRCSET).

References

[1] E. Petajan, B. Bischoff, D. Bodoff, and N. M. Brooke, "An improved automatic lipreading system to enhance speech recognition," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '88)*, pp. 19–25, ACM, New York, NY, USA, 1988.

[2] M. Brand and V. Kettnaker, "Discovery and segmentation of activities in video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 844–851, 2000.

[3] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[4] J. Yamoto, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov mode," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 379–385, Champaign, Ill, USA, June 1992.

[5] Y. A. Ivanov and A. F. Bobick, "Recognition of visual activities and interactions by stochastic parsing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 852–872, 2000.

[6] M. Petkovic, Z. Zivkovic, and W. Jonker, "Recognizing strokes in tennis videos using hidden Markov models," in *Proceedings of the IEEE International Conference on Visualization, Imaging and Image Processing*, Marbella, Spain, September 2001.

[7] E. Kijak, L. Oisel, and P. Gros, "Hierarchical structure analysis of sport videos using HMMs," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '03)*, vol. 2, pp. 1025–1028, September 2003.

[8] I. Kolonias, W. Christmas, and J. Kittler, "Automatic evolution tracking for tennis matches using an HMM-based architecture," in *Proceedings of the 14th IEEE Machine Learning for Signal Processing Workshop*, pp. 615–624, September-October 2004.

[9] P. Chang, M. Han, and Y. Gong, "Extract highlights from baseball game video with hidden Markov models," in *Proceedings*

of the IEEE International Conference on Image Processing, vol. 1, pp. 609–612, Rochester, NY, USA, September 2002.

[10] J. Assfalg, M. Bertini, A. D. Bimbo, W. Nunziati, and P. Pala, "Soccer highlights detection and recognition using HMMs," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '02)*, pp. 825–828, Lusanne, Switzerland, June 2002.

[11] N. Rea, R. Dahyot, and A. Kokaram, "Modelling high level structures in sports with motion driven HMMs," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, Montreal, Canada, May 2004.

[12] T. Starner, J. Weaver, and A. Pentland, "Real-time american sign language recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1371–1375, 1998.

[13] J. Hu, M. K. Brown, and W. Turin, "HMM based online handwriting recognition," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 1039–1045, 1996.

[14] G. I. Chiou and J.-N. Hwang, "Lipreading from color video," *IEEE Transactions on Image Processing*, vol. 6, no. 8, pp. 1192–1195, 1997.

[15] P. Morguet and M. Lang, "An integral stochastic approach to image sequence segmentation and classification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '98)*, vol. 5, pp. 2705–2708, Seattle, Wash, USA, May 1998.

[16] J. Boreczky and L. Wilcox, "A hidden Markov model framework for video segmentation using audio and image features," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 3741–3744, Seattle, Wash, USA, May 1998.

[17] N. P. Cuntoor, B. Yegnanarayana, and R. Chellappa, "Activity modeling using event probability sequences," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 594–607, 2008.

[18] P. Peursum, H. H. Bui, S. Venkatesh, and G. West, "Robust recognition and segmentation of human actions using HMMs with missing observations," *EURASIP Journal on Applied Signal Processing*, vol. 2005, no. 13, pp. 2110–2126, 2005.

[19] N. Robertson and I. Reid, "A general method for human activity recognition in video," *Computer Vision and Image Understanding*, vol. 104, no. 2-3, pp. 232–248, 2006.

[20] S. Reiter, B. Schuller, and G. Rigoll, "Segmentation and recognition of meeting events using a two-layered HMM and a combined MLP-HMM approach," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '06)*, pp. 953–956, Toronto, Canada, July 2006.

[21] T. Mori, Y. Nejigane, M. Shimosaka, Y. Segawa, T. Harada, and T. Sato, "Online recognition and segmentation for time-series motion with HMM and conceptual relation of actions," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3864–3870, Edmonton, Canada, August 2005.

[22] A. Kokaram, E. Doyle, D. Lennon, L. Joyeux, and R. Fuller, "Motion based parsing for video from observational psychology," in *Multimedia Content Analysis, Management, and Retrieval*, vol. 6073 of *Proceedings of SPIE*, San Jose, Calif, USA, January 2006.

[23] K. Holt, *Child Development: Diagnosis and Assessment*, Butterworth-Heinemann, Oxford, UK, 1991.

[24] S. Goddard, *A Teachers Window into a Child's Mind, a Non-Invasive Approach to Solving Learning and Behaviour Problems*, Fern Ridge Press, Eugene, Ore, USA, 1996.

[25] 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," *The Lancet*, vol. 355, no. 9203, pp. 537–541, 2000.

[26] E. Doyle, *Evaluation of movement programmes in the treatment of dyslexia*, Ph.D. dissertation, Trinity College, University of Dublin, Dublin, Ireland, 2008.

[27] S. J. Young, N. H. Russell, and J. H. S. Thornton, "Token passing: a simple conceptual model for connected speech recognition systems," Tech. Rep. CUED/F-INFENG/TR38, Cambridge University Engineering Department, Cambridge, UK, 1989.

[28] S. Furui, "Cepstral analysis technique for automatic speaker verification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 2, pp. 254–272, 1981.

[29] Y. Wang, Z. Liu, and J.-C. Huang, "Multimedia content analysis-using both audio and visual clues," *IEEE Signal Processing Magazine*, vol. 17, no. 6, pp. 12–36, 2000.

[30] P. Goh and E.-J. Holden, "Dynamic fingerspelling recognition using geometric and motion features," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 2741–2744, Atlanta, Ga, USA, October 2006.

[31] H. Wang, M. C. Leu, and C. Oz, "American sign language recognition using multi-dimensional hhidden Markov models," *Journal of Information Science and Engineering*, vol. 22, no. 5, pp. 1109–1123, 2006.

[32] B. Lehane, N. E. O'Connor, H. Lee, and A. F. Smeaton, "Indexing of fictional video content for event detection and summarisation," *EURASIP Journal on Image and Video Processing*, vol. 2007, Article ID 14615, 15 pages, 2007.

[33] L. Joyeux, E. Doyle, H. Denman, et al., "Content based access for a massive database of human observation video," in *Proceedings of the 6th ACM SIGMM International Workshop on Multimedia Information Retrieval*, pp. 46–52, October 2004.

[34] A. Kokaram, *Motion Picture Restoration: Digital Algorithms for Artefact Suppression in Degraded Motion Picture Film and Video*, Springer, Berlin, Germany, 1998.

[35] K. Y. Wong and C. L. Yip, "Fast rotation center identification methods for video sequences," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '05)*, pp. 289–292, Amsterdam, The Netherlands, July 2005.

[36] V. Luc and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583–598, 1991.

[37] F. I. Bashir, A. A. Khokhar, and D. Schonfeld, "Object trajectory-based activity classification and recognition using hidden Markov models," *IEEE Transactions on Image Processing*, vol. 16, no. 7, pp. 1912–1919, 2007.

[38] N. Liu, B. C. Lovell, P. J. Kootsookos, and R. I. A. Davis, "Understanding HMM training for video gesture recognition," in *Proceedings of the IEEE Region 10 Annual International Conference (TENCON '04)*, vol. 1, pp. 567–570, November 2004.

[39] S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland, *The HTK Book*, 1995, http://htk.eng.cam.ac.uk/docs/docs.shtml.