



Terms and Conditions of Use of Digitised Theses from Trinity College Library Dublin

Copyright statement

All material supplied by Trinity College Library is protected by copyright (under the Copyright and Related Rights Act, 2000 as amended) and other relevant Intellectual Property Rights. By accessing and using a Digitised Thesis from Trinity College Library you acknowledge that all Intellectual Property Rights in any Works supplied are the sole and exclusive property of the copyright and/or other IPR holder. Specific copyright holders may not be explicitly identified. Use of materials from other sources within a thesis should not be construed as a claim over them.

A non-exclusive, non-transferable licence is hereby granted to those using or reproducing, in whole or in part, the material for valid purposes, providing the copyright owners are acknowledged using the normal conventions. Where specific permission to use material is required, this is identified and such permission must be sought from the copyright holder or agency cited.

Liability statement

By using a Digitised Thesis, I accept that Trinity College Dublin bears no legal responsibility for the accuracy, legality or comprehensiveness of materials contained within the thesis, and that Trinity College Dublin accepts no liability for indirect, consequential, or incidental, damages or losses arising from use of the thesis for whatever reason. Information located in a thesis may be subject to specific use constraints, details of which may not be explicitly described. It is the responsibility of potential and actual users to be aware of such constraints and to abide by them. By making use of material from a digitised thesis, you accept these copyright and disclaimer provisions. Where it is brought to the attention of Trinity College Library that there may be a breach of copyright or other restraint, it is the policy to withdraw or take down access to a thesis while the issue is being resolved.

Access Agreement

By using a Digitised Thesis from Trinity College Library you are bound by the following Terms & Conditions. Please read them carefully.

I have read and I understand the following statement: All material supplied via a Digitised Thesis from Trinity College Library is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of a thesis is not permitted, except that material may be duplicated by you for your research use or for educational purposes in electronic or print form providing the copyright owners are acknowledged using the normal conventions. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone. This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

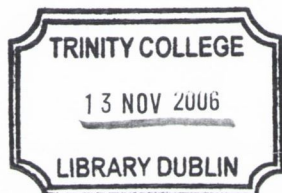
High-level Event Detection in Broadcast Sports Video

A dissertation submitted to the University of Dublin
for the degree of Doctor of Philosophy

Niall Rea
Trinity College Dublin, April 2005

SIGNAL PROCESSING AND MEDIA APPLICATIONS GROUP
DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING
TRINITY COLLEGE DUBLIN





THESIS
8021

To my family and friends

Abstract

This thesis investigates semantic analysis of broadcast sports footage. A domain dependent sports video model is proposed. Under this model, the game semantics can be derived according to their relationship with the sequence of dynamic events that occur in the sport and the evolution of the spatio-temporal behaviour of a relevant object. Snooker and tennis are targeted as typical broadcast sports footage for the purpose of this research. The problem focus is to automatically extract semantically meaningful events and to convey a useful representation to the user.

Access to semantics provides a more natural tool for a user to query a corpus of data than by low-level content based features alone. These semantics are however open to various interpretations by different viewers. Therefore, in order to create a successful semantic based retrieval system it is necessary to consider the user-context. Unconstrained sports footage is generally very complicated in structure, so restricting the domain being addressed enables a viewer model to be created. Domain specific features are extracted from the raw footage. These can then be exploited to develop algorithms which understand the characteristics of the data and the requirements of the user. These algorithms enable low-level domain features to be mapped to high-level semantics by learning the evolution of the features.

Traditionally, low-level visual features have been used to summarise the content in view. Global colour, texture and motion have all been used for this purpose. In this thesis a novel algorithm is presented which captures the geometry of the scene without having to extract and reconstruct complicated 3D scene geometry. Hidden Markov models are then used in a novel fashion to model these observations for camera view classification.

A new extension of the colour based Particle Filter is employed to track objects. It encourages better tracking in a constrained sports environment by exploiting prior scene geometry and playing surface colour information. The implementation of the tracker also allows for object collision and disappearance to be detected. The performance of the tracker is assessed using geometrical measures and by comparing it to the tracking produced using a gradient based motion estimator.

Thus far, retrieval of semantic events from sports footage has relied on prior knowledge of the broadcast video syntax. Typically, the temporal interleaving of camera views has been used to infer these important events or highlights in the footage. In this research, the spatio-temporal behaviour of tennis players and snooker balls are considered as being the embodiment of a semantic event. This concept offers a new means of automatically extracting semantic episodes from sports footage.

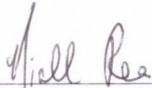
The scope of this thesis could easily be extended by further investigation into retrieval of semantic events from the vast quantities of other live and archived sports footage.

Declaration

I hereby declare that this thesis has not been submitted as an exercise for a degree at this or any other University and that it is entirely my own work.

I agree that the Library may lend or copy this thesis upon request.

Signed,



Niall Rea

April 28, 2005

Acknowledgments

To set these acknowledgements in motion I'd like to begin by thanking former, present and "touring" members of the Sigmedia group. Over the past three years they have provided me with great support in an engaging working environment. Each has chipped in to the completion of this thesis both directly and in round about fashion and their contributions can not be understated. I'd also like to thank the accountants of Enterprise Ireland, Trinity College Dublin and MOUMIR who managed to find the funding for my studies and travel.

A ridiculously massive amount of thanks must be heaped upon two very important members of Sigmedia; my co-supervisor Dr. Rozenn Dahyot and my supervisor Dr. Anil Kokaram. It is a testament to their work and guidance that this thesis arrived at its current state. I'd also like to thank everyone in the MEE department for their support for the past three years.

Finally I would like to express my indelible gratitude to my family and friends for everything.

Contents

Contents	iv
List of Acronyms	viii
1 Introduction	1
2 Visual Information Retrieval: A Review	6
2.1 Information Retrieval	7
2.2 Textual Annotation	9
2.3 Content Based Retrieval	10
2.3.1 Content Based Video Retrieval	11
2.3.2 CBVR Systems	12
2.4 The Semantic Gap	13
2.5 Sports Video Analysis	14
2.5.1 Temporal Structure Analysis	14
2.5.2 Feature Extraction	17
2.5.3 Event Detection and Recognition	21
2.5.4 Summarisation	27
2.5.5 Indexing	30
2.6 Overview of a Framework for Sports Video Analysis	31
2.6.1 Extraction	31
2.6.2 Recognition using Hidden Markov Models	33
2.7 Summary	35
3 Choosing Features for Sports Retrieval	37
3.1 The reasons for exploiting geometrical and colour content	38
3.2 Playing area segmentation	39
3.2.1 Segmentation using direct thresholding of colour spaces	39
3.2.2 Segmentation using adaptive thresholding of colour spaces	42
3.2.3 Colour space modelling	44
3.2.4 Choice of segmentation method	49

3.3	Playing area detection and inference of geometry	51
3.3.1	Geometry of a snooker table	52
3.3.2	Geometry of a tennis court	54
3.3.3	Radon Moment	62
3.3.4	Statistical colour and geometrical moments	64
3.4	Temporal boundary detection	68
3.5	Summary	71
4	Object Tracking	73
4.1	Probabilistic tracking	74
4.2	Particle Filtering	75
4.3	Generic implementation of the tracker	77
4.3.1	Establishing the likelihood model	78
4.3.2	Establishing the proposal distribution	81
4.4	Tracking snooker balls	85
4.4.1	Localisation of the white ball	85
4.4.2	Tracking the balls	88
4.5	Implementation of tennis player tracker	89
4.5.1	Localisation of the player	90
4.5.2	Tracking the player	91
4.6	Assessment of the performance of the particle filter	93
4.6.1	Perpendicular distance from points	93
4.6.2	Angle between least squares fit and true trajectory	94
4.6.3	Comments on the tracking performance	95
4.7	Tracking comparison	98
4.7.1	Gradient Based Motion Estimation	98
4.7.2	Quantitative comparison of tracking performance - GBME vs PF	99
4.8	Summary	99
5	Dynamic Event Detection in Snooker	103
5.1	Establishing initial ball colour models	104
5.2	Collision Detection	105
5.2.1	Dealing with shape distortion	106
5.2.2	Determining the colour of the new ball	108
5.3	Pot Detection	109
5.4	Foul detection	109
5.5	Snooker escape	112
5.6	Summary	113
6	Event Modelling and Classification using HMMs	114

6.1	Creating the Alphabet	116
6.1.1	Clustering using the K-means algorithm	116
6.1.2	Clustering using Gaussian mixture models	117
6.2	Hidden Markov Models	120
6.3	Defining a HMM	122
6.3.1	HMM topology	124
6.4	View classification of snooker and tennis sequences	125
6.4.1	View classification using the Radon moment feature	126
6.4.2	View classification using statistical colour and shape moments	128
6.4.3	Comments on classification and improvements by merging the results	129
6.5	Event Classification	135
6.5.1	Spatial encoding of the playing area	137
6.5.2	Parsing the footage at an event level	138
6.5.3	Model training based on human understanding of the events	142
6.5.4	Establishing the model topologies	144
6.5.5	Event classification results for snooker and tennis	151
6.6	Summarisation and Indexing	156
6.7	Summary	157
7	Conclusions and Future Directions	159
7.1	Future work	160
A	Hidden Markov Models	164
A.1	Issue 1: Evaluating the observation likelihood	165
A.1.1	Derivation of the forward variable.	166
A.2	Issue 2: Calculation of optimal state sequence and most likely state	167
A.2.1	Locally optimal state selection	168
A.2.2	Derivation of the backward variable	169
A.2.3	Derivation of the optimal state at time t	170
A.2.4	Viterbi algorithm: Most likely path via Dynamic programming	171
A.3	Issue 3: HMM parameter estimation using Baum-Welch	172
A.3.1	Baum-Welch algorithm	172
A.3.2	Training and recognition using scaling of forward and backward variables	173
B	A brief history of Snooker and Tennis, the basic rules and terminology	175
B.1	A brief history of Snooker	175
B.2	The basic rules - Snooker	176
B.3	Snooker Terminology	176
B.4	The basic rules - Tennis	178
B.5	Notes on the captured broadcast footage	179

C Particle Filter	181
D The Radon Transform	184
E Results of snooker and tennis view classification	186
E.1 View Classification using features individually	187
E.2 View Classification by cascading the classifiers	189
Bibliography	192

List of Acronyms

AR	A uto R egressive
CBVR	C ontent B ased V ideo R etrieval
DFD	D isplaced F rame D ifference
DHMM	D iscrete H idden M arkov M odel
DTV	D igital T ele V ision
GBME	G radient B ased M otion E stimation
GMM	G aussian M ixture M odel
HMM	H idden M arkov M odel
IR	I nformation R etrieval
MAP	M aximum A P osteriori
MMSE	M inimum M ean S quared E rror
MPEG	M otion P icture E xpert G roup
PF	P article F ilter
VIR	V isual I nformation R etrieval
VQ	V ector Q uantisation

1

Introduction

Research interest in high-level content based video analysis has grown in recent years [9, 25]. A good deal of this has been focused on the detection of semantic events that occur in sports video footage [5, 24, 31, 40, 48, 76, 172]. There are two primary factors contributing to this upsurge in research.

The Commercial Aspect: In a recent study performed for the European Union it was shown that sport themed channels are those showing the most considerable growth across the member States [111]. The commercial value of certain sports broadcast on these channels ¹ and the increasing choice being made available to viewers with the advent of Digital Television (DTV), has motivated broadcasting companies into finding additional means of exploiting the data set, from which to add to the marketability of the product. Interactivity is still hoped to be the killer application in new digital satellite and terrestrial services. The ability to choose the camera angle with which to view a soccer game, or being able to select a particular match from several concurrent games in tennis are just two examples of current trends in sports broadcasting.

The vast quantities of live and archived sports video material has resulted in demands by broadcasters for systems that ease the burden of annotating these bodies of data. This has motivated many researchers into undertaking the problem of high-level content based analysis of sports videos. This annotation process is currently a manual

¹In 2003 for example, a £1.024bn three year deal was struck by BSKYB with the English Football Association for the live broadcast rights to the Premier League [61]. While in 1998, the National Football League (NFL) agreed an eight year deal with four broadcasters worth \$18bn [53].

undertaking, where humans are responsible for accounting for the events which take place [9]. The existing manually derived metadata can be augmented by way of automatically derived low-level content based features such as colour, shape, motion and texture [41]. This enables queries against visual content as well as textual searches against the predefined annotations allowing for more subjective queries to be posed.

The User Consideration: Sports, as a genre, appeals to the general public and while most viewers will be content to view an entire game, some may only wish to view the highlights or a brief description of the events that occurred in the game. Alternatively, the user could specify the level of summary required (*e.g.* the entire tennis game with adverts, only the tennis, playing time or important events). This kind of freedom of access to the media has been made possible by DTV. It enables the viewer to effectively become the editor of their own programming with the content having been pre-recorded on a set-top box similar to the TiVo² or SKY⁺³.

The enhanced viewing abilities offered by DTV also offers a wealth of information to be made available to the viewer in the form of textual headers and content based descriptors. In the 2004 Six Nations Rugby, for example, the BBC provided statistics of previous games between the nations along with the player line-ups and textual meta data of the important events. The BBC coverage of the 2004 Olympics allowed the viewer to select one of 5 video streams of different events in conjunction with access to medal tables and a news ticker updating the user on the Games proceedings. Providing the user with the capacity to query broadcast footage at a high-level of abstraction to retrieve relevant events is a main area of research in many institutes.

Retrieval is a non-trivial task in general and is made even more difficult by the so-called “semantic gap” that exists between machine and user. As semantic level queries provides the most natural means for a user to query a corpus of data, it makes sense to develop algorithms that understand the nature of the data in this way. The typical user would rather search for this type of content using high-level queries rather than making use of low-level content descriptors. Integrating several low-level features can allow a user to search for high-level events but it is invariably cumbersome and time consuming. For goal events in a soccer game for example, a user would rather pose a semantic query (*e.g.* “Show me all the goals in the game”) rather than specifying low-level content based features such as percentages of dominant colour, velocities of objects, camera motion and the amplitude of the audio.

Semantics are subjective, so in order to create a successful retrieval system based on the semantics of the visual document, it is necessary to understand the user context. As the viewer operates at high levels of abstraction, semantic video indexing and domain specific video indexing are required. For example, inferring an important event in a sports game will

²TiVo: <http://www.tivo.com>

³SKY⁺: <http://www.sky.com/skyplus/>

require a different set of features to those needed to retrieve an important event in a talk show. This type of indexing can be accomplished by restricting the domain being addressed. These constraints enable low-level content based features to be mapped to high-level semantics through the application of certain domain rules.

The necessity for automatic summary generation methods is highlighted by the fact that the semantic value of sports footage spans short durations at irregular intervals during an event (high energy, short term episodes). A single day of test cricket can last six hours while a single frame of snooker will normally exceed 10 minutes. Interesting events occur intermittently, so it makes sense to parse the footage at an event level (where the event is related to a semantic episode). In cricket for example, an interesting event might be the bowler run-up, batsman's stroke and the direction of travel of the ball [82] while semantic episodes such as snookers, shot-to-nothings and break-building occur in snooker (please refer to Appendix B.3 for some snooker terminology). Considering the client or user end, a snooker game could be recorded on a digital set top box with integrated hard disk drive. The user could query the footage at a high-level of abstraction and the machine would return the relevant events from the video stream, perhaps with derived textual information giving the time at which the event happened, the player involved and a brief description of the event.

Content adaptation and automatic summary and index generation could also prove useful in the transmission of sports footage to low-bandwidth devices. For example, a 3G provider in the UK, ³ ⁴, offers their customers a sports service which transmits clips of English premier-ship soccer games direct to the user's 3G handsets. An automatic method of generating these clips (in the form of key frames or video skims), or different kinds of summaries which might be too tedious to be generated by hand (*e.g.* a cartoonised version of the event), could prove invaluable to this service. Furthermore, techniques to adapt the content to fit the display of a particular media device might also be needed. Broadcast sport footage contains shots where important events are most likely to be found pooled with replays, close-ups and crowd shots. Close up views often contain little semantic information relating to the events in hand and normally take place after an important event has happened. As there is generally no need to transmit these views, a significant amount of bandwidth could be freed up for relevant information to be transmitted. Automatically derived metadata from the broadcast footage could be added to the transmission in the form of closed captions, augmenting the description of the event. Backward compatibility could allow an SMS (Short Messaging Service) message to be sent to GSM compliant hand held devices or a picture message with added text and audio to 2.5G mobile phones.

This thesis concerns itself with retrieval and summarisation of semantic events that occur in broadcast snooker and tennis footage. It is arranged in 7 chapters of which, chapters 3, 4, 5 and 6 present the main contributions of the research.

⁴3: <http://www.three.co.uk/indexcompany.omp>

Chapter 2: Visual Information Retrieval: A Review

A review of the literature in the area of visual information retrieval is presented in this chapter. A framework for sport video analysis is discussed which involves temporal structure analysis, feature extraction, event recognition, summarisation and indexing of the footage. A review of research in the area of semantic based retrieval in sports is presented along with other areas which employ similar methods for different domains. The chapter concludes with an overview of the framework for semantic analysis of broadcast sports footage. The individual steps in the framework are considered in modular form under the headings of extraction and recognition. A high-level summary of each module is then presented.

Chapter 3: Choosing Features for Sports Retrieval

Common to any retrieval system is a feature extraction stage. This chapter details a new algorithm for parsing sports video footage. Based on summarising the geometrical content in view, the algorithm does not require the calculation of complex three dimensional scene geometry. Further features include the statistical moments of colour and geometrical image content, and their relevance to the parsing of sports video footage is discussed. A robust playing area detector for tennis and snooker, based on the Radon transform of a segmented colour space is also established.

Chapter 4: Object Tracking

In this chapter, a colour based particle filter based on the CONDENSATION algorithm [70] is outlined. Novel extensions of the trackers proposed by Perez et al [117] and Nummiaro et al [107] are used to encourage better tracking of objects in the sports domain. The implementation of the particle filter allows for the tracking of snooker balls and tennis players. The tracking results generated are then assessed using geometrical measures and compared to the tracks produced by a gradient based motion estimator for broadcast snooker footage.

Chapter 5: Dynamic Event Detection in Snooker

Dynamic events in snooker are important in so far as they affect the viewer's perception of the state of the game, allowing a rich set of semantics to be inferred. Methods which exploit the explicit motion tracks generated by the particle filter are used to detect dynamic events that occur in broadcast snooker footage. Three events are considered: ball pots, inter-ball collisions and ball-cushion collisions.

Chapter 6: Event Modelling and Classification using HMMs

In this chapter, special consideration is given to modelling the temporal evolution of low-level image features with a Hidden Markov Model (HMM). The modelling power of the

HMM enables it to cope with wide deviations in observation behaviour and create a signal model for each camera view.

Following correct labelling of each of the views, the concept of parsing sports footage at an event level is established. The evolution of the spatio-temporal position of a fundamental object in the footage is considered to embody the semantics of an event. The explicit motion tracks generated by the particle filter are quantised and a HMM for each event is trained based on a *human* perception of events in terms of the spatio-temporal feature. Finally, results of the event classification for snooker and tennis are discussed.

Chapter 7: Discussion and Further Research

In the final chapter of the thesis, the contributions of the research are assessed and ideas for future work are presented which might guide subsequent investigations into semantic analysis of broadcast sports footage.

2

Visual Information Retrieval: A Review

The concept of Visual Information Retrieval (VIR) encompasses the tools and methods used to retrieve data relevant to a query from large databases and archives. Queries can be made using either low-level visual content based features from images and video such as colour, texture, shape, *etc.* or high-level semantic content; objects, events and emotions for instance. This chapter addresses VIR from its inception in Information Retrieval through to its use in retrieval of semantic events from broadcast footage.

The need for VIR is becoming increasingly important with the wealth and speed with which visual information is being made available on digital media and in digital archives. The emergence of multimedia on the Net and the ease with which visual data can be distributed through high bandwidth transmission channels has highlighted the need for user friendly and efficient means of retrieval. The Getty image archive ¹ for example, contains in excess of 30 million unconstrained images while the BBC footage library ² contains more than two million subject listings on over 500 million feet of film and 400,000 hours of video. Cheap digital cameras and camcorders have enabled the home user to create personal image and video archives of several gigabytes in size.

The advances that have been made in techniques for VIR have however, been unable to match the level at which these visual documents are being produced. The unstructured nature of these ever expanding databases highlights a requirement for a cheap and efficient means of describing, retrieving and managing the vast quantities of audio-visual data.

¹Getty Archive: <http://creative.gettyimages.com/source/home/home.asp>

²BBC Footage Library: <http://www.bbcfootage.com/>

VIR has its roots in Information Retrieval (IR). In section 2.1, the concept of IR is introduced. A brief description of a traditional model used in IR is also presented along with a discussion on how wide-ranging the problem of IR is.

One means of achieving structure in video and image databases is by way of indexing. The established indexing technique of annotating visual documents is based around the traditional library paradigm. Manual generation of low- and high-level content descriptors are created by expert annotators in the form of textual metadata appended to the visual document. Textual annotation and its limitations are discussed in section 2.2. During the past decade, automatic derivation of high-level and low-level content based descriptors and the implementation of appropriate methods has been an area of much debate. The introduction of MPEG-7 has standardised the processes for the representation of multimedia content. In section 2.5.5 a brief description of the MPEG-7 standard is given along with the merits of the scheme.

Content Based Retrieval (CBR) has been used to complement existing retrieval methods where the stored metadata (usually in the form of textual annotations) is augmented by the incorporation of content based visual information. Current trends and techniques in Content Based Video Retrieval (CBVR) will be reviewed in section 2.3. While low-level content based information can be useful for some queries, it can not entirely be relied upon as being related to the semantic substance of the document. The so-called “semantic gap” that exists between machine and user will be discussed in section 2.4.

Semantic-level indexing of multimedia documents has a high expressive power and it can be used to describe most important aspects of the content. This form of indexing involves extracting the high-level content directly from the footage. The indexing is generally tailored to a specific domain [3, 5, 76, 157]. In section 2.5.3, a review of the literature in the area of semantic based retrieval from broadcast sports video is presented. Cognition based systems and hand written character recognition use similar techniques to those employed in high-level content video retrieval and these are also reviewed.

2.1 Information Retrieval

Information Retrieval (IR) was a term coined in the 1950's by Calvin Moores. He described it in [103] as a method that “embraces the intellectual aspects of the description of information and its specification for search, and also whatever systems, techniques, or machines that are employed to carry out the operation.” There are four key issues in IR as follows:

1. **Media Content Analysis:** This stage in IR relates to the information content of the document as perceived by the machine. This low-level machine perception of the media contains no semantically meaningful information that might be useful to the user.
2. **Pattern Recognition:** Secondly, the structure of these low-level features is analysed and the best fitting clusters are calculated according to a model. This model will

depend on the level of supervision (supervised/unsupervised) provided to the clustering algorithm (*i.e.* if the desired number of outputs is known *a-priori*).

3. **Relevance Feedback:** This stage of IR entails introducing human subjectivity in the form of relevance feedback [152]. As each document is open to different semantic interpretations by various users, the retrieval processes in such systems are often augmented by allowing human evaluation of the retrieval. This technique allows the user to weight the results based on their own perceptions to enhance the retrieval effectiveness for future queries.
4. **Evaluation of the retrieval:** A large part of IR involves evaluating the retrieval [72, 130]. Measures of the retrieval sensitivity involve assessing the relevance of the retrieval relative to a ground truth. Precision and recall are traditional metrics used in Visual Information Retrieval (VIR) systems. Relevance feedback also offers a way of measuring the effectiveness of the retrieval, where the measure is based on the subjective opinion of the user.

A simple, traditional model of an IR system is illustrated in figure 2.1.

The problem of successful IR is wide-ranging and extends from the retrieval of text documents using keyword queries to the retrieval of semantic events from vast archives of video libraries using high-level queries. An example of an effective IR system for the retrieval of scientific documents was proposed by Lawrence et al [86] through the NEC project Cite-seer³. It has proved to be an invaluable resource for the worldwide scientific community by allowing the user to retrieve scientific literature spanning the web. Citation indices are generated automatically along with abstract extraction and the provision of links to related and overlapping documents.

The concept of IR is straightforward: a user queries a database with the hope of retrieving information relevant to that query. A system that responds to a query in this fashion however is affected by user subjectivity, a central issue in any retrieval system. Relevance feedback is often used to circumvent this problem. It is often more appropriate however to extract additional features from the media document providing the user with more descriptive power. VIR systems can improve on the retrieval by incorporating automatically derived visual features (section 2.3) and semantics (section 2.5.3).

Colombo et al [27] refer to textual annotation systems and low-level content based systems as the first and second generations of visual information retrieval systems. While the first generation allows semantic based queries, the notions of the user must correspond to those of the annotator. As video and images are generally rich in high-level content the query could prove to be beyond that of the stored metadata. Even though the retrieval process is automated (*e.g.* extracting colour) in second generation systems it is generally not

³Citeseer: <http://citeseer.nj.nec.com/cs>

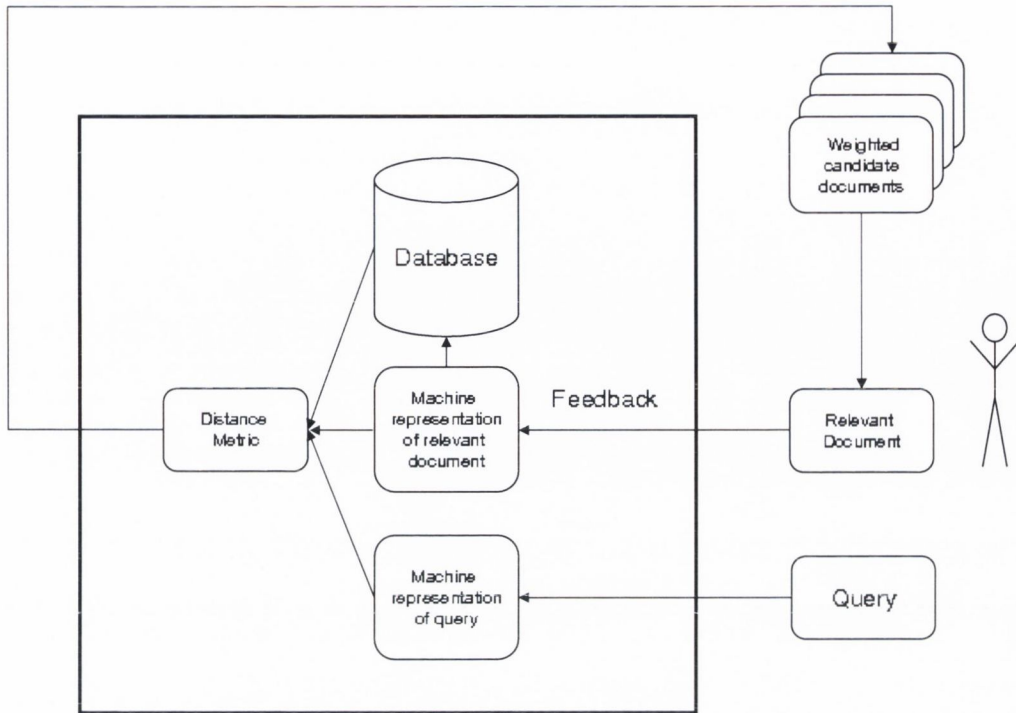


Figure 2.1: *Information Retrieval system with relevance feedback. The user selects an initial query and the system returns a set of documents. The user weights the relevant documents according to their perception. The distance metric and database are updated with this new information to enhance subsequent queries.*

possible to link low-level content to high-level concepts for unconstrained media. It is only through restricting the domain being addressed that this link can be established. Sections 2.2 and 2.3 discuss both generations of visual information retrieval and section 2.4 discusses the importance of domain restriction.

2.2 Textual Annotation

Comprehensive textual annotation systems have been in use for many years now. This process is currently the most direct, efficient and accurate means of finding “unconstrained” images and video in large unstructured databases such as the Web (for example the Google Image search ⁴). These systems are however subject to high costs as the annotations can only be obtained by manual effort. Transcripts, captions, embedded text, surrounding text and hyperlinked document type annotation are often used to represent the high-level concepts of

⁴Google Image Search: <http://www.google.com/imghp?hl=en&tab=wi&q=>

the images.

The use of textual retrieval in large image databases can be illustrated by considering the implementation of such a system in the Bridgeman Art Library (BAL) ⁵. BAL has built up an expert textual database on 750,000 of their images. Attached to each image is a manually entered set of metadata describing the image semantics along with its size, the name of the artist and several keywords describing the main content [135]. In such systems the search process is based purely on predefined attributes and the perceptions of the annotator. Such a retrieval system is open to the problem of user subjectivity due to the nature of the content rich images.

Recent increases in the computational power of PCs have allowed the use of previously inefficient language understanding algorithms to add textual information to multimedia documents. Computer assisted annotations in the form of closed captions can be added to the video for example. Keywords or keyphrases can be extracted from the audio track using techniques such as Term Frequency-Inverse Document Frequency (TF-IDF) [137]. TF-IDF uses Automatic Speech Recognition (ASR) to relate the frequency of words which occur in the audio stream to those stored in a database. Words which occur frequently in a segment but which are not as common in the database are assigned a high weight. It is assumed that these words are related to the visual content. These ‘important’ words can be utilised in the indexing of the documents allowing the user to pose text based queries and retrieve audio-visual clips which are synchronised with the derived textual metadata.

Compilation of metadata describing every aspect of a media document is unfeasible. Demand for an efficient and cheap means of retrieval is driving research in the area of CBR which improves on the limitations of traditional text based systems by providing extended access to the media.

2.3 Content Based Retrieval

CBR has received a vast amount of interest since the 1995 publication of the first paper [54] which formally addressed the subject by offering the potential to automate the retrieval process from visual media. CBR is a difficult problem. Extracting salient features from an image or video stream requires much more sophisticated methods than those used for parsing text documents for keywords [151]. Methods built around content analysis [106, 146], object segmentation [155] in images and shot classification [33, 68] in video have been developed to enable quicker and more efficient access to image and video documents.

Initially restricted to use in large image databases (*e.g.* The State Hermitage Museum ⁶ in St. Petersburg, Russia) the system provided good retrieval results for relevant content based queries. At the time, content based retrieval for video purposes was deterred due

⁵Bridgeman Art Library: <http://www.bridgeman.co.uk/>

⁶State Hermitage Museum St. Petersburg: <http://www.hermitagemuseum.org/>

to the prohibitive cost of storage and slow access times. Further development and new research [67] allied with the lower cost of storage media and video capturing devices has allowed video repositories such as the ABC News Video Source ⁷ to make use of low-level visual content for retrieval. Low-level content based information has been fused with existing textual annotations in the Bridgeman Art Library. Retrieval effectiveness has shown to be increased using this system rather than the annotations alone [135].

Combined with the adoption of the MPEG-7 standard ⁸ (a standard for multimedia description outlined in section 2.5.5), CBR has emerged as a major field of research. Content in visual media can be considered to encompass two levels of abstraction.

1. **Low-level:** Low-level visual content is generally described using colour, texture, shape and motion. These content descriptors are typically easily extracted from images or video and are chosen due to their efficiency, robustness and perceptual similarities. Traditional methods for CBR involve vectorising the image. An image can then be represented as a feature set and similarities between images can be measured by calculating a distance between these feature vectors. There are a number of different distance measures that can be used (*e.g.* Euclidean, Chebychev ⁹, Manhattan ¹⁰) for comparing feature vectors, but none have been established as a definitive model for human similarity evaluation.
2. **High-level:** High-level content is embodied by both semantic and affective-content information [16]. The semantics relate to an event or object in the event, whereas the affective-content is the reaction triggered by that same semantic event. For example, in soccer, if a goal is scored, the semantic event is the goal itself. The effect of the goal (the affective content) is the player celebration and crowd reaction.

High-level features are more difficult to extract from the media than low-level as they are open to different interpretations by viewers. Techniques have been formulated which typically rely on restricting the retrieval to a unique domain and then mapping the low-level features to high-level concepts by modelling the temporal evolution of the low-level visual features. It was concluded by Roach et al [129], that narrowing the domain being addressed is a good means of bridging the semantic gap.

2.3.1 Content Based Video Retrieval

Content based video retrieval (CBVR) is a burgeoning area of CBR. The main goal of CBVR is the full automation of the process of parsing, indexing and describing low- and high-level content using multimodal information.

⁷ABC News Video Source: <http://www.abcnewsvsources.com/vsource/html/home.htm>

⁸MPEG7 ISO/IEC JTC1/SC29 WG11: <http://ipsi.fhg.de/delite/Projects/MPEG7/>

⁹Chebychev distance: <http://www.comp.lancs.ac.uk/kristof/research/notes/basicstats/>

¹⁰Manhattan distance: <http://mathworld.wolfram.com/TaxicabMetric.html>

2.3.2 CBVR Systems

Over the past number of years CBVR systems have become available by commercial vendors and academic institutes. Initially, the focus of much research was on adapting existing CBIR systems for video retrieval purposes. Consequently, this meant that the inherent audio and motion information were not exploited, and the temporal aspect was eliminated by only considering keyframes from automatically detected shots. A summary of some existing systems is given below.

- **QBIC**¹¹: The QBIC (Query By Image Content) system [54] was created by IBM as a means of retrieving images from large databases. The system enables operators to query the database using a pictorial example or sketch which can comprise a combination of shape, texture, colour and spatial location descriptors. The system then calculates a distance metric between the query and the corpus and returns images which minimise the distance. A ranking can then be performed based on the distance. The main advantage of this system is that it allows the user to query the database without using text. The query is based solely on low-level image content which eliminates individual user interpretations. The image based QBIC system was adapted [95] to allow queries against a video database by incorporating a shot cut detector to extract a keyframe for each shot. This effectively eliminates the defining temporal feature of video by generating a “storyboard representation” of the entire video. The task now becomes one of image comparison and thus ignores any evolution of the descriptors which may occur over the duration of a shot.
- **Informedia**¹²: The school of Computer Science at Carnegie Mellon University has developed a content based video retrieval system giving users access to over 1500 hours of news and documentaries from which a number of papers have been published [62, 63, 123, 173]. The Informedia project (started in 1994) attempts to facilitate machine understanding of video to allow efficient summarisation and retrieval of relevant content. The video is indexed using automatically transcribed audio tracks, closed captions or extracted on-screen text. Text based queries are then compared to the precomputed indices to retrieve visual summaries from the vast corpus. Approaches toward multimodal queries are currently being considered which would allow all features of the video medium to be exploited.
- **VisualGREP**¹³ : The VisualGREP Project at Mannheim University employs a domain-independent search by video sample technique to retrieve video footage of varying durations using features including colour, motion and object types. The features are

¹¹QBIC: <http://www.qbic.almaden.ibm.com/>

¹²Informedia: <http://www.informedia.cs.cmu.edu/>

¹³VisualGREP: <http://www.informatik.uni-mannheim.de/informatik/pi4/projects/MoCA/Project-visualGREP.html>

combined by the user, where (s)he weights the features according to the required query. The MoCA project, discussed in section 2.5.4 is one application of the VisualGREP framework.

- **VideoQ**¹⁴: The VideoQ system developed at Columbia University is a web based video retrieval system which allows the user to formulate a query by means of an animated sketch. The chief difference between this system and the others is that it incorporates the spatio-temporal information of objects into the query. The user can specify the colour, shape and texture attributes of objects along with the required trajectories in the video. Video objects in the original footage are spatially and temporally segmented off-line using a combination of edge, colour and motion continuity information and shot cut detection respectively. The segmented object characteristics are then approximated using colour, texture, shape and motion features. The similarity between the query and the corpus is calculated using a composite distance comprised of a user specified weighting of each of the attributes in the query. A keyframe from each of the candidate clips is returned.
- **Físchlár**¹⁵: Developed in Dublin City University, the Físchlár system allows registered users on the local area network to record broadcast television programmes from eight channels. The system parses the video and extracts relevant keyframes using the method outlined in section 2.5.4 enabling the user to peruse the video using a web browser or mobile device.

2.4 The Semantic Gap

The most natural means for a user to query a corpus of data is by way of semantics. As CBVR systems operate in terms of low-level or primitive visual features, they have no concept of the semantics of images or video clips. Even though low-level content can sometimes be related to high-level semantics, the machine cannot perceive it as such. For example, if a user wishes to find all the morbid pictures in a database, and can query by colour content, he will probably pose the query with a substantial amount of black, and other dark colours. To the machine, the operator is simply looking for images with low-values of luminance. Overcoming the semantic bottleneck by enabling high-level understanding has been an area of much research in both CBIR and CBVR systems [50].

The fundamental problem with semantics in general, is that they are open to an individual's own interpretation. This is referred to as semantic ambiguity in O'Leary [110]. In other words, human judgement is conditioned by intuition, experience and expertise. It is not feasible to assume that a retrieval process could be created that would be able to understand

¹⁴VideoQ: <http://www.ctr.columbia.edu/videoq/>

¹⁵Físchlár: <http://www.cdvp.dcu.ie/>

the complex human thought process which would allow for high-level semantic queries in broad domains. On the contrary, in current systems, humans have to translate the semantic contents into low-level descriptors in order to find an appropriate document. The retrieval is then based on the assumption that the semantics of the document are correlated with the visual content, which is not always the case.

The extraction of semantics and translation of low-level to high-level content is still an open issue, and there has been no unilateral resolution on how to accomplish this. The most common approach used for facilitating semantic queries has been by tailoring the retrieval to a unique domain [3, 5, 31, 39, 59, 76, 125, 141, 157]. This has achieved success in the sports domain (in tennis [74] and baseball [24] where the temporal interleaving of camera views was noticed to exemplify semantic events for example) and in other broadcast programming such as wildlife videos [59] where the presence of certain motion patterns is used to indicate the occurrence of hunts in the footage.

2.5 Sports Video Analysis

The work presented in this thesis is restricted to the sports domain, specifically to snooker and tennis. The goal then is to create models which exploit low-level features and are able to retrieve semantic events which occur in broadcast footage. Our work was one of the first to approach broadcast sports footage for this purpose. A five stage systematic approach to sports video content analysis is presented in this section. The stages involved are:

1. Temporal structure analysis.
2. Feature extraction.
3. Event detection and recognition.
4. Summarisation.
5. Indexing of the footage.

Each stage in the process is described in sections 2.5.1-2.5.5 and a discussion of the methods employed in other works is presented. The focus of the reviews are not solely on sports video processing as much of the research in generic video content analysis is applicable to that used in sports footage.

2.5.1 Temporal Structure Analysis

A video can be organised by analysing the relationship between its temporal segments which comprises of a hierarchy of frames, shots and events for sports or scenes for non-sport video (figure 2.2 illustrates the hierarchy of a typical video sequence). The first step in uncovering the temporal structure of the video involves the detection of temporal boundaries.

The problem of shot cut detection in any video footage is considered to be generic, so the following section unifies the various sport and non-sport approaches. Detection of gradual shot transitions proves to be more difficult than the problem of shot cut detection. In the subsequent section, some techniques dedicated to the detection of gradual effects will be reviewed.

Once temporal boundaries have been established the location of shots are known. A shot is considered to be the basic logical unit of a video which is delimited by the locations of the temporal boundaries. A shot can therefore be defined as a sequence of contiguous frames which is continuously captured by a single camera.

Shot Cut Detection

Shot cut detection techniques exploit the inherent relative homogeneity of frames in a shot in terms of their colour and motion content. Hence, a large variation in the correlation between consecutive frames indicates the presence of a shot cut. A variety of features have been exploited to good effect to characterise this homogeneity; the sum of histogram differences [14], edge pixel enumeration [167] and MPEG DCT coefficients [170].

It was noted in Ekin et al [48] that there is a high correlation of colour content present in different camera views in some sports footage. This is due for the most part to the colour homogeneity of large background playing regions (*e.g.* soccer or American-football type pitches). A three feature colour based approach was therefore proposed in [48]. It fuses the difference in colour histogram similarity with dominant colour pixel ratios in a particular frame and the difference between dominant colour pixel ratios of two frames under a robust classifier, which adapts based on the local content. Incorporating spatial information into shot cut detection, Tan et al [144], divide the DC-image of an MPEG encoded sequence into 12 rectangular regions. The intensity histogram of each region is computed and compared to that of the corresponding region in the successive frame. Most of the significant shot changes were found. A similar spatial segmentation of each frame is undertaken in Pickering et al [121]. In this research each frame is divided into 9 blocks. Shot cut detection is performed by calculating the Manhattan distance between the RGB colour distributions of each corresponding block in consecutive frames. Vasconcelos and Lippman [153] create a statistical framework for shot segmentation which incorporates prior shot duration knowledge into the decision process. Results of the method are compared to those achieved without a prior to illustrate the importance of considering temporal features for shot cut detection.

Due to the limitations of colour based approaches, several methods that make use of other features enabling shot cuts detection have been proposed. In Kokaram et al [82] for example, global motion estimation is applied to the detection of shot cuts during “action sequences” in cricket footage. As the camera cuts to a long framing view of the playing field when the bowler run up is followed by a hit, a noticeable discontinuity in the diagonal affine motion

transformation parameters is exhibited. This is due to the global motion changing from predominantly zoom to pan left or right. A shot cut and gradual shot transition detection method based around the tracking of feature points in texture, such as corner points, was proposed by Abdeljaoued et al [1]. Each of the feature points is tracked from frame-to-frame using a Kalman filter. The rate of change of disappearance of points, and emergence of new points was used to infer the type of transitions that occurred. Results show a significant improvement over standard histogram techniques.

Gradual Shot Transition Detection

Detection of gradual transitions is considerably more difficult than that of the shot cut detection problem. These types of production effects are broadcaster or event dependent and usually include variations in the types of wipes, dissolves and fades. Wipes may include a logo while the rate of dissolve might vary for different programs. Several robust algorithms for shot transition detection have been developed using statistical methods [38], pixelwise comparisons [158] and edge pixel information [168].

Wu et al [158] propose a solution to wipe detection in video using the DC-images in an MPEG encoded sequence. A wipe stripe which is evident in the pixelwise difference of consecutive I/P frames characterises the boundary between the two images. A statistical measure of the stripe enables wipes and camera motion to be differentiated. Similar to their previous paper [167], Zabih et al [168] propose a method for detecting a variety of production effects based on edge pixel enumeration and the spatial distribution of edges. The scheme is based around the fact that new and old edges appear and disappear far from each other assuming that the frames have been compensated for global motion.

In section 3.4 shot cuts and gradual shot transitions are detected by exploiting shape and spatial luminance correlations between consecutive frames of sports footage.

Temporal Hierarchy

While shots do not provide much insight as to the overall content of a video, they can prove to be useful as a unit for indexing a visual document. Combined with domain constraints, an understanding of important episodes can be derived. For example, in most sport applications, the main action takes place in a certain camera view. If this view can be categorised, relevant shots can be extracted from the sequence and labelled. Furthermore, the occurrence of shots in a particular order can point to certain high-level events [76]. Figure 2.2 illustrates the hierarchical structure of a video sequence.

It is important to distinguish between scenes and events in video. In this thesis we consider an event to be the basic high-level element during which an important episode takes place in sports footage. An event in a tennis match might be a rally or ace for instance. A review of techniques related to the detection of events in sports is described in section 2.5.3.

Scenes are defined as a group of shots with the same thematic content unified by space, time and event. For example, a scene of a conversation might comprise several shots of the people talking in a certain environment. So, a scene in a sport event rarely changes, except perhaps where there is a change from the playing arena to a studio for analysis of the game. Events within each scene in sports footage are therefore considered to be of most importance.

For completeness some techniques that enable the detection of scenes in video are reviewed. Detection of scenes in video is considerably more complicated than temporal boundary detection and normally involves the incorporation of prior domain knowledge. Background tracking techniques can be used to detect scene boundaries where the locale changes. In Schaffalitzky et al [132] salient points on rigid 3D objects are used to identify shots with the same background content using wide-baseline methods. The technique is invariant to the camera viewpoint, occlusion and object scaling. Background tracking is used to calculate scene cuts and compare the semantics of scenes in Oh et al [109]. A fixed background area is defined *a-priori* and a Gaussian pyramid is used to reduce its representation to a background signature. Two consecutive signatures are compared by shifting them in opposite directions. If a continuous match, less than the length of the signature is found, a scene cut is presumed to have occurred.

2.5.2 Feature Extraction

The first step in content based video analysis and processing involves identifying features in the footage that the user can exploit in order to formulate a query. Colour, shape, motion and texture have all been used to this effect. These features are chosen because they are generally easily understood by human operators and similarity measures can generally be easily computed. Since broadcast sports exhibit different patterns of such low-level content they prove to be useful for retrieval purposes. Work in extraction of these features is now discussed under the relevant headings.

Colour

Colour features are exploited in most retrieval systems [6, 33, 54, 115, 134, 136]. It is perhaps the easiest low-level feature for human operators to perceive and can, at the same time, be considered to give a good summary of the video or image content. For example, in image sequences with significant dominant colour, a single value can be used to summarise the image [33]. Colour features also offer scale invariance and are generally efficiently computable. Furthermore, techniques have been established for comparison of features such as histogram differencing and the Bhattacharyya distance [117]. From the human psychological point of view, colour properties of images are very useful in that they can sometimes be associated with the semantics in an image. A user could, for example, search a database for an image of a beach scene by specifying the quantity of yellow (sand), blue (sand and sea) and white

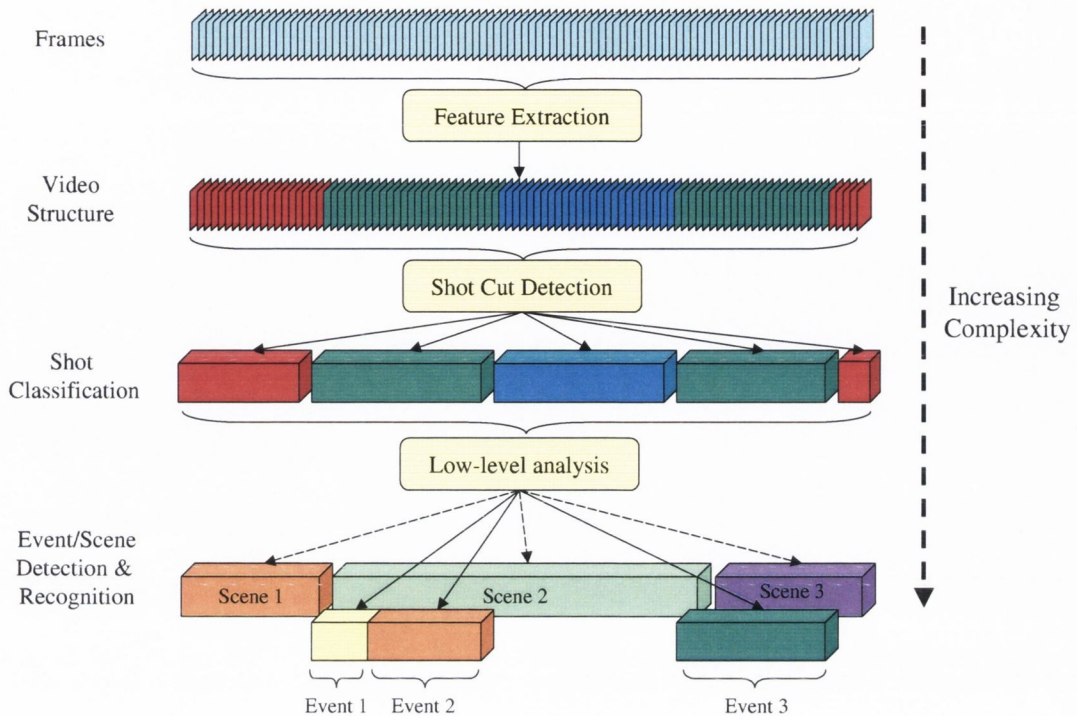


Figure 2.2: Hierarchical structure of a video. Relevant features are extracted from the footage which is then broken into its constituent shots. Low level analysis is then undertaken which pertains to some high-level processing on the low-level features. This allows high-level semantics to be inferred (these might be events for sports or scenes in generic video).

(clouds) present.

Colour histograms [14], colour coherent vectors [113] and the moments of colour features [33] have been used to describe low-level visual content in images and videos. Colour histograms are the most traditional and common means of expressing the colour properties of an image. Histograms are an approximation of the colour distribution in the image but do not account for the spatial arrangement of the colours in the image. This can be useful if there is a requirement for the query to be rotationally invariant but in the majority of cases a lack of spatial information will be detrimental. Colour correlograms incorporate colour and spatial information. They express how the spatial correlation of pairs of colours change with distance [66]. Colour queries are formulated in query by sketch systems by selecting a colour for a particular region from a predefined palette. The system then retrieve images or video that best match the chosen colour.

In the sports domain, dominant colour regions have been used for the detection of playing surfaces [24, 48, 74, 141]. Ekin et al [48] developed a colour region detection algorithm which

automatically detects the colour of the sports field and adapts to spatio-temporal variations in the dominant colour. In Chang et al [24] ratios of grass and sand along with other shape descriptors are used for classifying different views in baseball. Jain et al [141] calculate the most frequent colour in a specified region for classification of a tennis court surface. The distance between it and the mean value of a trained set of predefined colours is then calculated. The result of minimum distance is set as the appropriate playing surface (*i.e.* clay, grass, hard and carpet type tennis courts). Zivkovic et al [175] take a different approach by modelling the colour properties of a tennis court using a 3D Gaussian distribution in RGB colour space. It is assumed that the colour distribution is unimodal due to the high occurrence of playing surface pixels in the full-court view. Only a single Gaussian is therefore required to model the colour distribution. However, while not explicitly stated, it seems that only an indoor playing surface is considered. Furthermore, only footage from one source is used. Due to the hard court surface, it does not experience any degradation in surface quality. The court is therefore not subject to any changes in colour as is the case for grass and clay. MPEG-7 colour descriptors have also been used for the retrieval of high-level events in sports footage [64]. In section 3.3.4 we exploit the colour content exhibited in different camera views for the purpose of view classification.

Motion

Extracting motion information provides another feature essential for video content analysis. The inherent temporality of video is manifest through camera motion or the motion of objects in the scene. The motion of objects of interest means that motion as a feature is key in any video analysis. The intensity of camera motion or object motion can be evaluated using techniques such as motion estimation [81] and edge change ratios (ECR) [167] or by extracting existing motion vectors from an MPEG encoded sequence [114].

Motion features provide access to rich semantics in the footage. For example, they can be used to identify the level of “action” in a sequence because high levels of action will usually be manifest as high intensity motion vectors [20,52]. Kokaram and Delacourt [82] exploit this observation in the sports domain where global motion is used to classify high-level events in cricket. A view of the bowler run up is signified by an increase in the zoom parameters, while a batsman stroke results in a change in the global motion translation parameter, the sign of which represents the direction of the hit.

Camera motion has been used for shot type classification in a number of publications relating to the sports domain. Assfalg et al [5] use motion features to describe the type of framing use in soccer footage as one of very long framing, long shot and medium shot. Chang et al [24] classify individual shots in baseball by noticing that those shots of the same type have comparable distributions of camera motion along with colour and texture. Gibert et al [159] extend the concept used in [24] to classify different sports footage as being one from

ice hockey, baseball, American football and soccer. They assume that different sports (and not individual shots within the footage) exhibit different motion patterns. Motion vectors extracted from MPEG encoded sports footage are used as content descriptors to classify the various individual sports.

For the most part, only camera related motion has been used for high-level retrieval in the sports domain. Explicit tracks of relevant objects have not yet been considered for this purpose. In section 6.5.2 a new means of detecting high-level events which occur in sports is presented. It exploits the fact that in certain sports, the behaviour of particular objects encapsulates the meaning of high-level events.

Shape

Much in the same way as machines used for computer vision decompose complex 3D objects into simpler volumetric components, it has been shown that the human visual system performs similar operations when attempting to analyse images [10]. This means that humans can instantly recognise objects by shape features alone. Querying by simple shape features can therefore be considered to be an effective method for retrieval.

Most shape descriptors rely heavily on good segmentation [15]. Following segmentation, object areas are labelled and spatial measurements such as area, centre of gravity and eccentricity are taken. A query is formed by computing the same features in the query visual document and computing a similarity metric against the corpus.

In the sports domain there has been much interest in describing the playing areas in terms of their geometrical content. Since sports playing surfaces are generally well defined in terms of their geometry, the arrangement of lines in a particular order can signify the presence of a certain camera view or the possible occurrence of a high-level event. Gong et al [58] attempt to locate all incidents around the goal, corners and open play in a soccer game by recognising the required arrangement of lines. Ekin et al [48] also deal with soccer by attempting to detect three parallel lines for the retrieval of the goal region. In Zivkovic et al [175], after the player is segmented from the tennis court standard shape features such as orientation and eccentricity along with centre of gravity, area and distances of the extrema from the centre based on a pie structure are used to characterise the player. Shape descriptors specified in an MPEG-7 stream are used by Höynck et al [64] to detect objects in equestrian footage. A highlight is deemed to have occurred upon detection of such an obstacle as a jump will have been attempted.

In sections 3.3 and 3.3.3 a novel shape descriptor for classifying camera views is presented. The descriptor does not require the computation of complicated 3 dimensional geometries. Four steps (segmentation, edge detection, Radon transform and moment calculation) are sufficient to describe the shape content of sports footage which exhibit strong geometrical content in terms of the shape of the playing area. This value can then be used as a shape

index for each frame. Another shape feature which exploits the alignment of local edges is also used to characterise the shape content of a frame.

Texture

The textures of regions in an image are characterised by variances in brightness levels. The texture of an object can therefore be considered as describing the relationship between adjacent pixels in an image. Texture, while not being particularly useful independently, can complement the use of other features where those features alone cannot sufficiently describe an object. For example, if two regions of similar colour properties border each other, a description of the textures will help in their disambiguation, for example a picture leaves and grass.

Texture descriptors may be computed using frequency techniques, such as wavelet decomposition [131]. These methods are based on relating the spatial arrangement of pixels to the degree of coarseness of the texture in the visual document. In sports, texture is used in conjunction with other features by Kittler et al [79] as a semantic cue for the presence of objects in broadcast sequences.

Similar to that of the colour feature, systems which allow queries based on texture use a predefined set of palette textures which the user selects for a particular region. The system returns images or video that best match the chosen texture.

2.5.3 Event Detection and Recognition

Following temporal and low-level content analysis, the semantic content can be extracted from the footage. In order to do so, the machine must understand the events in hand. Retrieval techniques in some of the systems discussed previously, apply to corpora of unconstrained images or video. Successful semantic level retrieval based on high-level queries on such bodies of unconstrained information are currently not possible as the retrieval system would have to understand all the information presented. So, in order to implement successful retrieval techniques based on semantic queries it is necessary to constrain the problem to a unique domain.

Robust techniques, which might be useful for user based on-line semantic level query applications, are essential in an age of emerging interactive television. A review of some of the methods used for semantic analysis of sport events is presented in subsequent sections. In most of these cases, analysis of the content is performed on observable low-level features, and probabilistic or deterministic models are subsequently used to classify the particular semantic events.

Later in this section, the field of emotion recognition is briefly reviewed. Cognition based systems are considered relevant for review as the problem of classification of content in these systems and semantic based retrieval are relatively similar tasks. Both problems involve au-

tomatically learning and classifying contents of the video sequence by analysing the temporal patterns of low-level features.

Event Recognition in Sports

Much research in retrieval has been focused on the detection of semantic events that occur in the sports domain. As individual sports tend to have different rules, it becomes necessary to further restrict the domain to a unique sporting event. While some research propose a generic solution to detection of events in any kind of sports footage [162, 172] an overlap in feature space could cause some events to be misclassified.

Thus far, techniques for the retrieval of important events in sports including soccer [5, 48, 58, 160], American football [93], baseball [24], tennis [33, 76, 172], snooker [39, 125, 126], cricket [82], basketball [144] and equestrian sports [64] have been sought. The problem has been approached using both unimodal [5, 40] and multimodal [31, 48] data.

The inherent temporal nature of video manifest by the evolution of video features typically shows wide variations in behaviour. Modelling these often inhomogeneous features is difficult and pointwise statistics do not suffice. For example, features which are subject to noise and behave impulsively such as those used in automatic speech recognition, a more complex model than, say a Gaussian needs to be used. This has lead to an increasing interest in the use of Hidden Markov Model (HMM) based classifiers [5, 24, 76].

The following section introduces the concept of the HMM and the discusses the motivation for its use in video sequences. The subsequent section discusses some publications which have exploited HMMs to model various semantic events that occur in broadcast sports footage.

Hidden Markov Models

This section will provide the reader with sufficient knowledge of the HMM to appreciate the concepts outlined in the literature review of the following sections. Chapter 6.2 and appendix A deal with HMMs in greater depth.

HMMs have been shown to be one of the most efficient tools for processing dynamic time-varying patterns. Their use has found considerable success in applications where these patterns are particularly evident, for example in cognition based systems and video processing applications. They allow a variety of temporal patterns to be modelled as the model topology can be chosen such that it reflects the nature of the data. Figure 2.3 shows the structure of a left-to-right first order HMM with N states. The left-to-right model has been found to well represent problems that are inherently temporal since the structure follows the nature of temporally evolving data (*e.g.* automatic speech recognition where each state or a number of states represents a word phone).

The model in figure 2.3 is first order in the sense that the current state relies only on the state that preceded it. Given a sequence of states $\{q_1 \dots q_t\}$, under a first order Markov

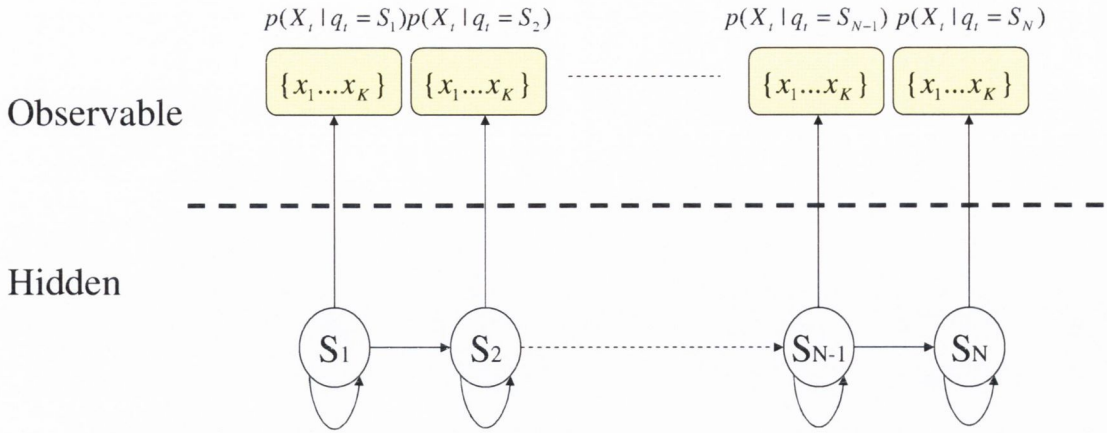


Figure 2.3: First order left-to-right Hidden Markov Model

assumption the probability of state q_{t+1} , can be written as:

$$p(q_{t+1}|q_t \dots q_1) = p(q_{t+1}|q_t) \quad (2.1)$$

In more general terms, a state-to-state transition is encoded by a transition probability matrix \mathbf{A} , where $q_t = S_i$ and $q_{t+1} = S_j$ are the realisations of state S_i and S_j at times t and $t + 1$. Equation 2.2 defines the state transition probability matrix.

$$\mathbf{A} = \{a_{ij}\} = \{p(q_{t+1} = S_j | q_t = S_i)\} \quad (2.2)$$

The model is initialised by specifying the probability of being in a particular state, π_i at the first time instance, $t = 1$, (*i.e.* $\pi_i = p(q_1 = S_i)$).

For each state in the HMM, an observation vector, $\mathbf{B} = b_j(x_k)$, is defined which may be continuous or discrete. Observations, ($V = \{x_1 \dots x_K\}$), are therefore a function of their state. Equation 2.3 is the observation emission probability mass function (pmf) (or pdf for the continuous case) associated with state j .

$$b_j(x_k) = p(X_t = x_k | q_t = S_j) \quad (2.3)$$

The hidden nature of the HMM means that only the observation pattern is seen and not the state sequence (the state sequence can however be derived from the observations with the Viterbi algorithm (section A.2.4)). A HMM can therefore be implemented to represent the statistical nature of the observations in terms of a network of states and for each observation the process occupies a particular state in the HMM (figure 2.3).

The parameters of the HMM are typically estimated from training with ground truth data. The algorithm used is called the Baum-Welch algorithm and is described in appendix A.3. The following two sections show how the HMM has been used for different applications in sports.

Event Recognition in Sports Video using Stylised Production Information

Due to practical limitations, there can only be a finite number of cameras mounted at fixed locations in the broadcast of any sport. These cameras transmit a continuous video stream to an editing suite. When dealing with broadcast footage, the coverage of some sports is typified by the stylised interleaving of these camera views interspersed with production effects such as dissolves and wipes. Some work in high-level content analysis exploit this inherent trait in the footage to classify semantic events [5, 24, 76].

To convey particular semantic episodes, most sports highlights are composed of a specific number of interleaved camera views with a certain temporal structure. This means that events can be detected by applying certain semantic constraints in terms of the video syntax. This characteristic is prevalent in what are known as action-stop sports such as tennis and baseball, where each semantic episode is punctuated by a period of non-action. Non-action events are typically communicated via a crowd or close-up shot of the player. All of the implementations that use HMMs in this way are variations on the same theme. The models are simply adapted in order to suit the appropriate domain (that of identifying semantic events through patterns of view). Similar techniques are used for parsing broadcast news footage [23, 46].

Kijak et al [76] deals with the mapping of the temporal structure of raw tennis broadcast footage to high-level concepts such as aces, rallies and service breaks using HMMs. Tennis footage has a particular video syntax which is exploited in this paper. A rally for example, can be modelled using a left-to-right model where one state is a non-global view (*i.e.* any view other than that of the full court - NV) and the other is a global view (*i.e.* a full court view - GV). In other words, in broadcast tennis footage a rally is typified by a full court view preceded by a non-global view such as a close-up of the player or crowd. Figure 2.4 shows the model for a tennis rally. A higher level HMM is then used to reflect the tennis game in terms of points. This is achieved by concatenating previous HMMs (*e.g.* a point is achieved when a first serve+rally or rally is followed by a replay).

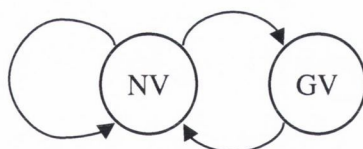


Figure 2.4: HMM for a tennis rally in [76] where NV is a non-global view and GV is a global view.

In a similar fashion to Kijak, Assfalg et al [5] identified that significant events in soccer (*e.g.* corner kicks, free kicks and penalty kicks) are almost always taken with long shot framing

(from a camera that is mounted on one of the stands and takes a wide view of the pitch) interspersed with medium (a view showing the player(s) and some of the pitch) and short shot framing (a close up). Due to the commercial nature of soccer, it is one sport into which considerable research has been invested [5, 48, 143, 148, 160] in an attempt to uncover patterns in the spatio-temporal dimensions thereby allowing semantic events to be inferred. The moving ball is considered to embody the hidden process and domain knowledge helps deduce the number of states required for the HMM. The values corresponding to camera action, representing the moving ball, are therefore considered to be the observations.

Chang et al [24] use HMMs to detect highlights in sports footage. Their techniques are restricted to the domain of baseball. Each semantic episode (*e.g.* home-run, good catch) is modelled on the video syntax. An architecture is determined by analysing the temporal and spatial domain specific structures, unique to the game. The observation vector used to drive the HMM comprises several visual features including motion, edge and playing area descriptors. The model that best fits the observation sequence is returned as the matching event. Likewise, Li et al [93] model plays in American football for coaching video analysis. Two algorithms are proposed, one deterministic (see subsequent section) and one probabilistic. A HMM models the views which had been classified using temporal and colour constraints. It was shown that the probabilistic approach achieved better classification results in 3 out of 4 footage sources and equal classification in the final source.

The retrieval process considered in Ekin et al [48] differs from the previous attempts of high-level event recognition in broadcast soccer videos. They instead rely on feature based deterministic methods to classify high-level events. Cinematic features are derived from the dominant colour content from which three different types of shot are classified (long shots, in-field medium shots and out of field/close-up shots. Goals are detected using a cinematic template composed of an interspersion of these types of shots, the existence of slow motion replays and the duration of the break in play when a goal has been scored. Object features (referee and penalty box detection) are used to discover higher-level events in the play. When a close-up of the referee is detected for example, he might be showing a red or yellow card to a player. When the penalty box is detected, an attempt on goal could be taking place. The retrieval framework is extended in [49] by creation of a generic integrated object-based video event description model. Event models are formulated by describing the event in terms of actors, interactions, motion (elementary motion units (EMU)) and reactions (elementary reaction units (ERU)) realised by the extraction of low-level descriptors.

Other methods of event Recognition in Sports

Modelling the temporal evolution of interspersing camera views has proved to be successful in retrieving high-level events that occur in a variety of sports footages. Statistical methods applied to other features have been shown to be equally effective in retrieving valuable high-

level content.

Successful semantic level retrieval has been performed by Petkovic et al [119] by limiting the search domain and taking advantage of HMMs. The paper addresses CBVR by recognising events in a tennis video using a Discrete Hidden Markov Model (DHMM). The model is driven by spatial features extracted from a binary map of player following its segmentation from the tennis court by a robust colour segmentation algorithm. A model is trained on observation features enabling high-level queries to be performed. A similar problem was undertaken by Yamoto et al [163], however broadcast tennis footage was not used. They presented an early paper in which HMMs were applied to a computer vision problem and is recognised as the first action recognition method using HMMs. A set of time sequential images of a tennis player is quantised into a discrete sequence of symbols. Mesh features are then used as the domain specific feature vector (the ratio of black pixels (background) in the binary image of the player to the total number of pixels). A more advanced feature set is used to accomplish a similar objective in Lee et al [87]. In this paper human actions (jumping, sitting, walking, *etc.*) are classified in close to real-time. Wu et al [157] attempt to classify different types of track and field events by analysing changes in global motion accelerations. A three level architecture of neural networks (NN), decision trees and finite state machines (FSM) is used to map low-level features to semantic episodes.

The importance of retrieval of semantic content is again highlighted in Xu et al [162]. In this paper, various levels of semantics in sports footage are represented by a corresponding layer in a multi-level HMM framework. The method used in this paper attempts to derive a generic solution to semantic based retrieval for sports (volleyball and basketball are analysed). At the lowest level, features based on motion are used to drive a HMM for each event in the particular sport. Each event is then represented by a state in the higher level HMM. The likelihood of each model is then calculated and the semantic is inferred.

Kawashima et al [73] attempt to summarise and index events in baseball using domain specific heuristics and multimodal techniques. A characteristic view from behind the pitcher is used as the basic scene from which the summarisation is begun. Detection of on-screen text allows the beginning of a batting sequence and change of player to be recognised. The text is then extracted and optical character recognition allows scores to be extracted. Batting actions can be detected in the basic scene by calculating a feature vector of the moving areas using frame differencing and comparing it to a model. The sequence is terminated when a hit or strike or ball has been detected. A hit is assumed if the view cuts to one where all the bases are in view, a strike or ball is assumed otherwise.

In [60], Hanjalic proposes a generic solution to event detection in broadcast sports footage. By fusing low-level multimodal features (motion activity, shot cut density and sound energy), exciting periods in the game are considered to have occurred. As this method is not confined to a single sport event, specific high-level events cannot be sought for in the footage. It is therefore only suitable for summary generation purposes and not retrieval. Audio features

alone have been used by Marlow et al [97] which offer good summary results for various sports. A mosaicking scheme for the summarisation of soccer footage is proposed by Yow et al [166]. Important events are detected by recognising frames which contain the goal posts and a panoramic image is constructed by compensating for global motion. A track of player movements and ball positions are overlaid on the mosaic providing an effective summary of an exciting segment.

Classification of content in cognition based systems

In many areas of video processing, modelling the dynamic behaviour of features is important. In cognition based systems for example, it was recognised that it is not essential to reconstruct complex human geometry and movement in an attempt to recognise human actions [26, 104, 140, 145]. This observation can be related to one of the problems in this thesis of modelling the motion of an object around a playing area.

HMMs allow the temporal nature of low-level features related to human movements to be modelled. It is a widely used modelling technique for gesture and handwriting recognition applications and has been used successfully since Starner et al [140] in 1995. Gesture recognition is undertaken in Cohen et al [26] in which they propose a multi-level HMM for the automatic segmentation and recognition of human facial expressions. On-line handwriting recognition has also been modelled by a HMM in [89] and [164].

2.5.4 Summarisation

Summary generation has been a main area of research in content based video analysis in recent years [48]. Summarisation involves locating and extracting important events and conveying the information to the user in a concise manner. Good summarisation is vital given the vast amount of data associated with any video document. This is particularly the case for sports where the most valuable semantics only occupy short time periods relative to the total duration of the footage.

Good summarisation of content can be provided by Motion History Images (MHI) [18] where a synthetic representation of object motion is overlaid on a keyframe. MHIs were not originally intended for summarisation purposes, but prove particularly useful for video where motion conveys some substantial semantic information. The motion history is collected by frame differencing and is overlaid on an average of the first and final images in the sequence. The motion is represented by a temporally graduating intensity which increases over time. Frames with recent motion are therefore represented by bright regions and earlier motion by darker regions. The first application of MHIs in sport can be found in Denman et al [39]. An illustration of MHIs for snooker is given in figure 2.5. In the presence of global motion the use of MHIs for summarisation is impractical unless the camera motion can be compensated.

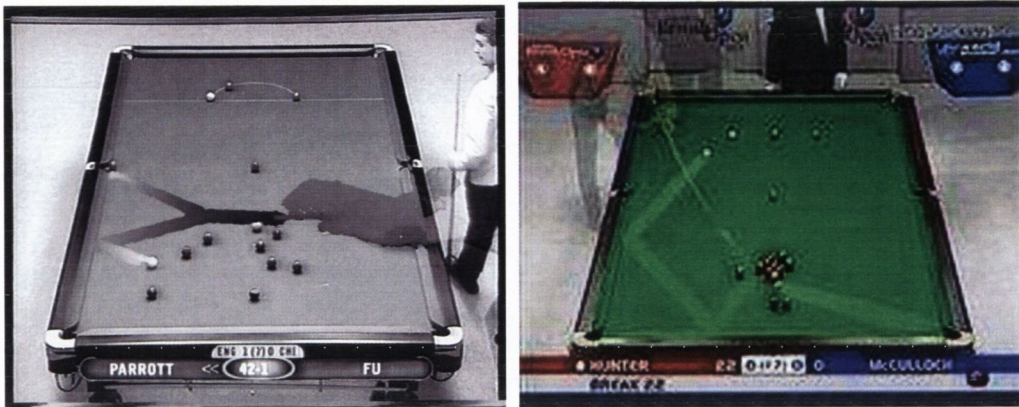


Figure 2.5: *MHI of snooker shots.*

The most common methods of summarising video are keyframes and video-skims. Reviews of works in the area are presented in the following sections.

Keyframes

A keyframe is a single frame extracted from video footage which is considered to give the best representation of the events by capturing the visual content of the shot. Keyframes provide a static representation of a dynamic event which enables a user to efficiently browse a corpus of video material or annotators to index footage. Keyframes are the most commonly used method of video summarisation. Chapter selection on DVD movies is one example of their use for commercial purposes. Below is a review of some of the techniques in this area.

Keyframes are typically extracted from each shot in the footage but several keyframes may be used to convey object or global motion [171] based on certain criteria. The keyframe selection process is generally based on a similarity metric between frames or a rule based approach. The Físchlár system¹⁶ implemented by the Center for Digital Video Processing at Dublin City University Ireland for example, uses a thumbnail keyframe browser where the keyframes are selected from the footage based on analysis of the colour content in each shot [88]. The colour distribution of a frame that is of closest distance to the mean colour histogram for the shot is selected as the keyframe.

Another keyframe extraction algorithm was presented by Liu et al [94]. Shots are segmented into motion events by using the Perceived Motion Energy (PME). The motion vectors from MPEG B-frames are used to calculate the average motion magnitude and direction of the motion, the product of which yields the PME. A triangular motion model is fitted to the PME representation to temporally segment the footage where the model represents motion

¹⁶Físchlár: <http://www.cdvp.dcu.ie/>

acceleration and deceleration. A heuristic rule based approach is used to extract keyframes according to either the detected positions of the initial acceleration and terminating deceleration or from frames extracted using the twin-comparison gradual transition effect detection method [169].

A seminal paper in adaptive keyframe extraction is described in Zhuang et al [174] which uses clustering methods for keyframe selection. The clustering, which is based on the K-means algorithm, is initialised by selecting the first frame in the shot and comparing it to consecutive frames. If the distance is less than a predefined threshold, it is deemed to be part of the cluster and the centroid is adjusted accordingly. A new cluster is instantiated if the similarity is less than another threshold. Frame similarity is calculated based on the distance between colour histograms. A cluster that is big enough is a *key cluster*, and the keyframe is the frame closest to the cluster centroid.

A comic book style summary is proposed by Boreczky et al [19] and Uchihashi et al [150]. As in [57], the video is clustered based on smoothed 3D YUV colour histograms. This produces clusters which are independent of their temporal attributes. When the clusters have been formed, continuous segments can be derived by seeing to which cluster each frame belongs. Keyframes in each segment are selected according to their importance in the footage which is calculated based on the duration and rarity of the segments. The weighting can also be adjusted based on the application. In this publication, more emphasis is put on those shots with humans present. For this to be applied to a sports problem, a greater weight could be applied to those shots where there is a greater probability of an important event occurring, such as the full court view in tennis or a side on view in basketball. Figure 2.6 shows a manually generated summary of how a tennis sequence could be summarised using this technique. Using the audio track to detect racquet hits [31], the global views could contain a motion summary for each pair of shots made by the players. A novel feature of the paper is the comic book style of the browsing interface. The size of the keyframes (calculated using an importance metric) reflects their importance in the footage.

Video skim

A video skim is a condensed audio-visual clip of a longer sequence, comprised of automatically extracted shorter clips which preserving the “message” from the original footage. In order to generate a perfect video skim, a high level of understanding of the footage is required and the footage semantics must firstly be derived to ensure that the best clips for describing the remainder of the footage are extracted. In other words, video skim generation is domain dependent and does not offer the same flexibility as keyframes for summary generation.

It has been shown in Smith et al [137], that video processing alone cannot be relied upon for generating a good video skim. Through the integration of image and language understanding, Smith et al create a coherent synopsis (in the region of 10:1 compaction ratio) of

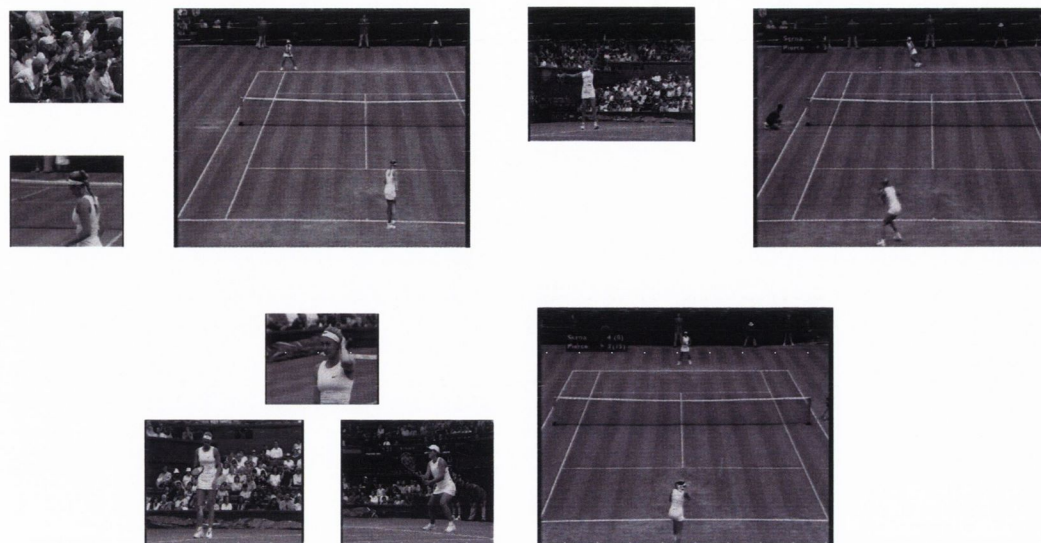


Figure 2.6: *Comic book summary of a tennis game.*

the original material. Keywords are detected in the audio track using Time Frequency Inverse Document Frequency (TF-IDF), and keyphrases are extracted using a predefined heuristics. Video skim candidates are established by classifying scenes using video processing techniques including shot cut detection, global motion analysis and object detection. Using high-level meta rules the temporal correlation of the skim candidates and extracted keyphrases allow for a video skim of the original video to be created.

The MoCA (Automatic Movie Content Analysis) Project [120] at the University of Mannheim has concentrated on the automatic abstraction of movies based on content analysis of the video. Heuristics are used to create a movie trailer where certain cinematic events such as action or dialogue are detected using the video and audio tracks to detect significant events. These scenes are concatenated to produce a movie abstract. No user evaluation of the trailers was presented so the performance of MoCA can not be assessed.

2.5.5 Indexing

Following the temporal and spatial segmentation of the multimedia document, it is indexed using the appropriate derived metadata. Depending on the type of document being indexed the metadata to be appended might comprise textual headers, visual and audio features or some other temporal information.

The MPEG-7 standard is now introduced which has stemmed from the worldwide requirements for the creation of a standard specifically designed for representing multimedia content. In 2001 the ISO finalised and approved the MPEG-7 standard. The primary aim of

MPEG-7 is to create a framework which is able to describe all the characteristics of multimedia documents using four elements: descriptors, description schemes, description definition language and coded descriptions. Low-level visual (such as those described in section 2.5.2) and audio features are contained in the descriptors while the descriptor schemes create a structure by relating the individual descriptors to one another.

The main advantage of MPEG-7 is the potential for interoperability between compliant devices for easy identification, retrieval and categorisation of multimedia documents. Searches for relevant documents will become more efficient as the feature descriptors will not have to be calculated for individual queries as they will already be present in the document.

2.6 Overview of a Framework for Sports Video Analysis

The review has shown that high-level event spotting in sports has been primarily based on the arrangement of particular view types. By restricting the domain to sports, this thesis proposes to shift the focus from these characteristics of sports footage, to objects contained in the footage. The behaviour of these objects help bridge the semantic gap.

The proposed framework for sports video analysis follows the steps outlined in section 2.5 and is illustrated in figure 2.7. The system is composed of two alternating module concepts: Extraction and Recognition. Extraction encompasses temporal analysis of the sequence, extraction of low-level features to yield correct classification of camera views and the extraction of motion features enabling high-level events to be inferred. The features are quantised as a symbol sequence which represents the observed views and semantics. The recognition module is a HMM driven by the symbol sequences and a maximum likelihood decision is employed for classification of camera views and high-level events.

2.6.1 Extraction

The fixed nature and finite number of cameras broadcasting a sport event means that domain-dependent information can be exploited to extract low-level features from the footage. In this research, three colour and four shape features are used to extract information relating to which view is being broadcast.

Another feature used is the motion of a fundamental object in the 'global' view. In this thesis the motion trajectories of the white ball and tennis player in snooker and tennis respectively are considered to embody a semantic event. A method for accurate tracking of the object is needed which enables abrupt changes in motion to be detected.

Choosing Features for Sports Retrieval

The first level in the framework is to extract relevant features for sports retrieval. Playing areas in broadcast sports footage are generally well defined in terms of their colour as well

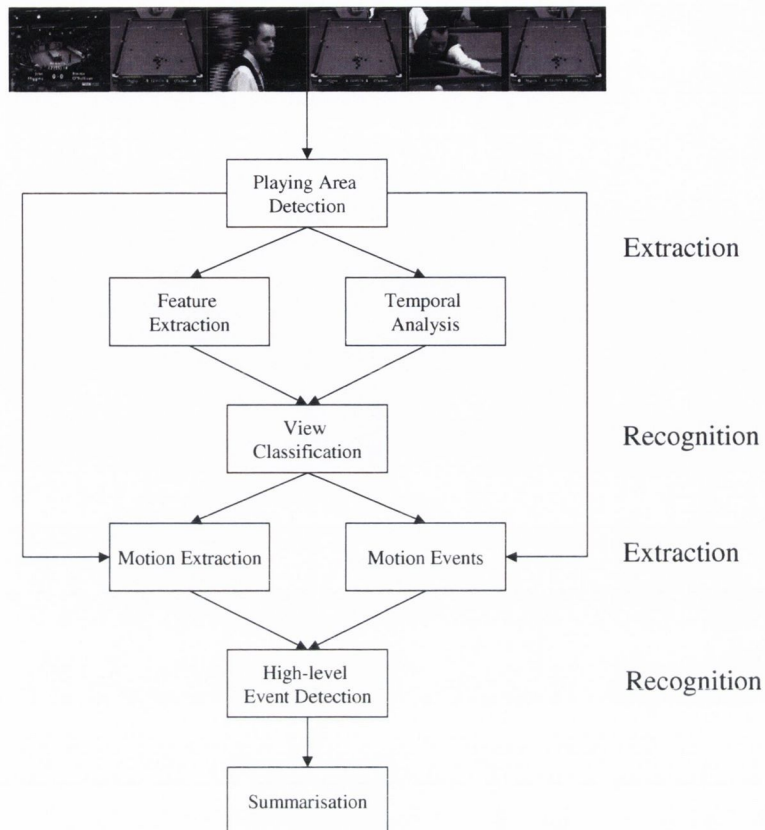


Figure 2.7: System for parsing broadcast sports footage.

as geometry. For instance, tennis can be played on green, red or blue surfaces with white delineating field lines in a rectangular shape. Snooker tables are green with a contrasting background colour. The playing surface shape is also rectangular. The low-level content based features used in this research encompass both of these playing surface attributes. The features used are easily and efficiently calculable. Novel moment features which provide a succinct single value representation of the image surface are used. Temporal analysis is performed by exploiting the extracted geometric features and established temporal boundary detection techniques.

The work reported here appears in the Journal of Computer Vision and Image Understanding: Special Issue on Video Retrieval and Summarisation as a paper entitled “Content based analysis for video from Snooker Broadcasts” by H. Denman, N. Rea and A. C. Kokaram [40] and was also published in the proceedings of the International Conference on Image and Video Retrieval as a paper under the same name [39]. This is the first stage in the framework illustrated in figure 2.7.

Motion Extraction and Motion Events

Extracting the motion of relevant objects is the fourth step in the framework (figure 2.7). Object tracking is performed using a colour based particle filter. The implementation differs from others [107,117] in that it exploits prior scene geometry and colour for better tracking fidelity. A target model of the object's colour distribution is created in the first appropriate frame of the footage. The likelihood of candidate models generated from particles distributed around the projected position of the region in the next frame are computed and weighted accordingly. If the cumulative likelihood of the samples is greater than a specific threshold, the mean location of the samples is taken as the location of the object. Explicit tracking of the object in this fashion also enables a summary of the event to be shown in terms of the temporally evolving position of the object overlaid on a keyframe from the footage.

Sudden changes in the behaviour of the object can be taken to indicate a change in perception of the event that might occur in the eyes of the viewer. In snooker footage for example, a cushion bounce that occurs before an inter-ball collision could indicate that the player is attempting to escape from a snooker (see appendix B for snooker terminology).

2.6.2 Recognition using Hidden Markov Models

The second and fourth levels in the framework are performed by the recognition module. Recognition is performed at both low- and high-levels of abstraction. Low-level analysis is performed by modelling the evolution of the moment features using HMMs for view classification.

View Recognition

As correct detection of the required views is essential to perform high-level content analysis, the stochastic nature of the moment features within each view is modeled using a HMM. Although the sequence is relatively homogeneous, it is subject to variations within each clip. This is due to subtle camera motion and the occlusion and uncovering of parts of the playing areas as a result of player movement. The use of a HMM fulfills the requirements of statistically modelling of the underlying signal. Figure 2.8 shows the evolution of the eight order central moment of the Radon transform for snooker footage. This appeared in a paper entitled "*Sport Video Shot Segmentation and Classification*" by R. Dahyot, N. Rea and A. C. Kokaram [33] in the Visual Communications and Image Processing conference.

Event Recognition

High-level content based analysis is achieved by performing object based analysis. If the spatio-temporal evolution of an object in view can be modelled using statistical processes, high-level semantics can be inferred from low-level observations. This approach can be con-

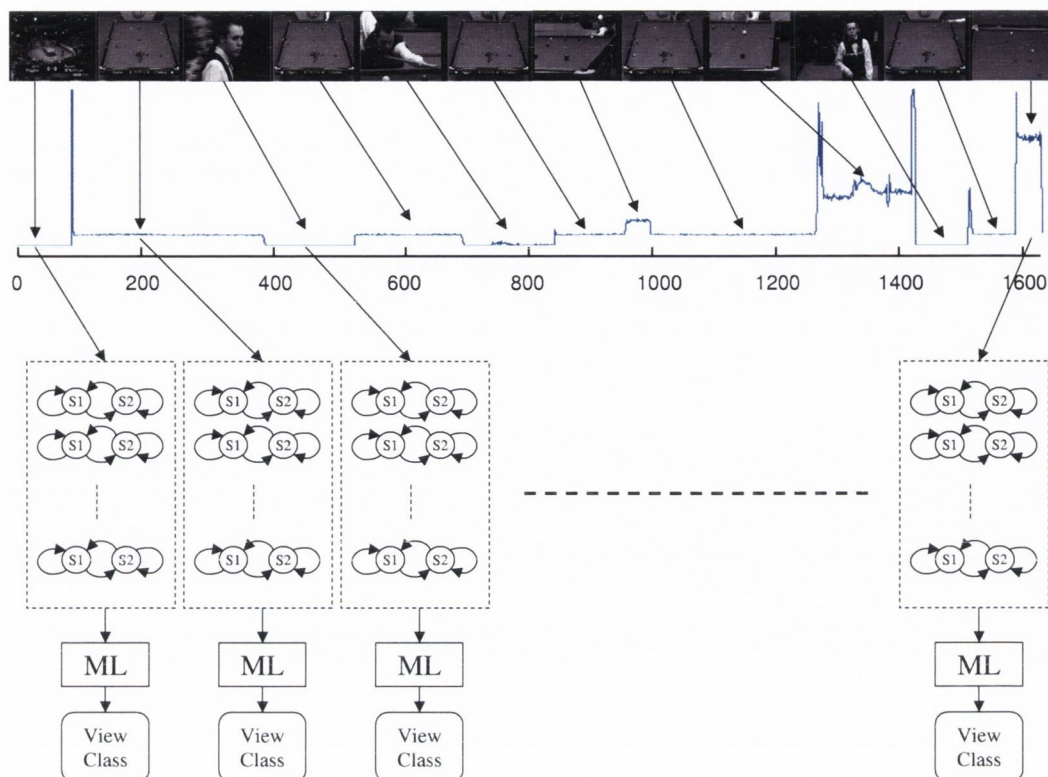


Figure 2.8: View recognition overview where *ML* is maximum likelihood classification.

sidered to be generic for sports where the movement of an object around a predefined playing area can be considered as being the embodiment of a semantic event.

In this thesis, snooker and tennis footage are used as examples of broadcast sports footage. For snooker footage, the motion of the white ball, and its interaction with other balls and the table is taken to symbolise certain events or plays that occur during the game. While in broadcast tennis footage, a model created of the track borne out by a tennis player in the lower half of the court, is used to ascertain the type of play under way. Other sports where this principle could be used might be table-tennis, squash and badminton, while tracking of a ball in soccer or rugby could elicit the appropriate semantics.

Having successfully classified the camera views and the compilation of the object tracking having been completed, the spatio-temporal evolution of these positions in terms of a spatially segmented playing area are modelled using a HMM. The topology of the HMM is derived from the data where each state is representative of a segment of the playing area. Continuing from the view recognition step in section 2.6.2, an overview of the event recognition is illustrated

below.

Event recognition was the theme of two papers entitled “*Modelling High Level Structure in Sports with Motion Driven HMMs*” by N. Rea and R. Dahyot and A. Kokaram [125] and “*Semantic Event Detection in Sports through Motion Understanding*” by N. Rea, R. Dahyot and A. Kokaram [126] appearing in the IEEE International Conference on Acoustics, Speech, and Signal Processing and in the 3rd International Conference on Image and Video Retrieval respectively.

2.7 Summary

As the quantity and range of content increases so has the need for the number of means to effectively mine it. This chapter has presented a review of the literature under the heading of the steps in a proposed framework for sports video analysis. The review has shown that the approach that most of these researchers adopt begins with a low-level feature extraction stage. The low-level features are then processed and subsequent high-level reasoning is applied in order to detect high-level semantics.

The remainder of the thesis will be considered under two headings: Feature Extraction and Recognition. Feature extraction will deal with the first two problems in the framework of temporal structure analysis and feature extraction from the raw video footage. Recognition will engage the problem of view classification, event detection and recognition along with summarisation and indexing. The architecture of the full system for parsing snooker footage is illustrated in figure 2.9. Views are classified and followed by event classification. A similar system is adopted for tennis footage analysis.

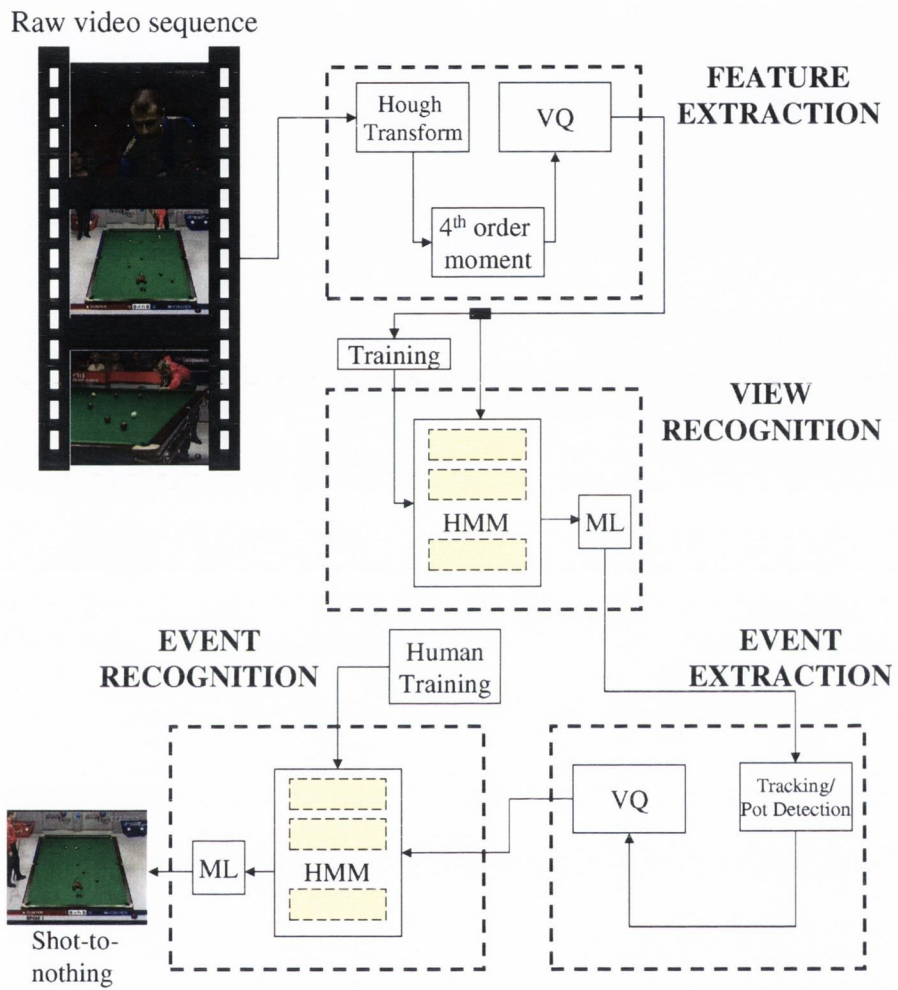


Figure 2.9: Overview of a system for parsing snooker footage and detecting events where VQ is vector quantisation.

3

Choosing Features for Sports Retrieval ¹

In section 2.6, a proposal for a five stage framework for sports video analysis was presented. This chapter details the steps involved in the implementation of the first two stages used for analysis of broadcast sports footage: feature extraction (section 3.2-3.3) and temporal structure analysis (section 3.4).

For the purpose of this research, two sports are considered, one of which is played indoors (snooker) and the other outdoors (tennis) ². These sports were chosen as they both exhibit strong geometrical content in terms of their playing areas. Furthermore, they are very structured in terms of their unambiguous rule sets.

Feature extraction is central to any successful retrieval system. The features used must give a good representation of the content while being efficiently calculable. The feature extraction stage is in this case begun by finding the delineating field lines or playing surface boundaries for a particular sport. Once these lines have been detected, other important regions on the surface such as the pockets and spots in snooker and the net and service boxes in tennis can be inferred using the initial playing area outline.

In section 3.3.3, a new feature for parsing sports footage is presented. Based on the strong geometries of the delineating playing area, this feature does not require the calculation of complex three dimensional basis sets to model the camera view as in Jain et al [141]. Instead,

¹Results from this chapter have been published as “Content based analysis for video from snooker broadcasts” by H. Denman, N. Rea, and A. C. Kokaram in the *Journal of Computer Vision and Image Understanding (CVIU): Special Issue on Video Retrieval and Summarisation*, November-December 2003.

²Although it is appreciated that tennis can be played indoors, the footage used is that from outdoor grass court and clay court tournaments.

it relies on clearly delineated field lines or contrasting playing surface and background colours to label the appropriate view type.

Statistical moments of colour and shape features are also considered for classification of the different camera views. The shape feature differs to that of the Radon moment in that all the information in the scene is used by measuring the alignment of the local edges. The colour feature is a single value representation of the colour distribution in the image. Using colour information for sports footage is appropriate as the different camera views used to capture the footage will exhibit different colour content from view-to-view.

By modelling the temporal structure of the evolving features, it will be shown that high-level events can be detected. As a first step toward exposing this temporal structure, the boundaries between homogeneous feature behaviours should be detected. These boundaries can be detected by computing a correlation measure between successive frames. There are many means of detecting shot boundaries [92]. In this work three steps are used. Initial shot boundaries are detected using scene geometry from the feature extraction stage. The remaining shot cuts are detected using a traditional absolute sum of luminance histogram differences. Other gradual transitions such as dissolves and fades are detected using a modification of the technique outlined in Zhang et al [169].

3.1 The reasons for exploiting geometrical and colour content

Broadcast snooker and tennis footage exhibit many similar characteristics to most other televised sports. The finite number of fixed camera views are arranged in such a way as to cause the viewer to become immersed in the footage while trying to convey the excitement of the game to a mass audience. The most important views in the footage can be considered to be those of the full table in snooker and the full court view in tennis. These type of views have been dubbed “global views” [76], the remaining views are “non-global”. Amongst others, these include close-ups of the player, crowd and other views generally not used for live broadcast of action events.

In snooker, global views are considered to be of most importance as they hold the fundamental details about the state of the game. All ball positions and pockets can be retrieved from the footage using this camera view, while practically all of the player’s shots and ‘pots’ are normally shown in this primary view. From the three footage sources of snooker used as data material in this thesis, an average of 63.92% of the total footage duration is spent in the full table view. Table 3.1 illustrates the time spent in the global view relative to the total footage length, along with the number of shots and unique camera views used in the broadcast.

Similarly, in the tennis footage a considerable amount of the total duration of the footage is spent in the global view. Over four different games, 47.56% of the total duration is spent in this view. The most important events in a tennis game are also deemed to occur in the

Footage	<i>Hunter</i>	<i>Hendry</i>	<i>Higgins</i>
# Frames (Total)	24250	5832	3491
# Shots (Total)	115	21	23
# Unique views	14	5	6
# Frames (Full View)	16323	2894	2243
% Full view duration	67.31%	49.62%	64.25%

Table 3.1: Table showing the ‘value’ of the full table view in terms of broadcast time occupied by this single view.

global view. Rallies, aces and other shots are initially broadcast in this view, while other non-global views of the court are generally used in replays. The importance of the global view for each of the footage sources is shown in table 3.2.

3.2 Playing area segmentation

To correctly segment the snooker and tennis playing surfaces from the background, three commonly used methods were implemented: direct thresholding, adaptive thresholding and colour distribution modelling. For each of these segmentation methods, different colour spaces are used. Utilisation of the colour spaces is based on the perception in quality of segmentation achieved. Further details about colour spaces can be found in [14] and [56].

3.2.1 Segmentation using direct thresholding of colour spaces

Direct thresholding of colour spaces has been used to good effect for segmentation purposes [14]. Given that an image or sequence of images will generally exhibit peaks at certain

Footage	<i>Pierce</i>	<i>Hewitt</i>	<i>Malisse</i>	<i>Costa</i>
# Frames (Total)	2949	12009	4114	11000
# Shots (Total)	16	59	18	75
# Unique views	5	9	7	7
# Frames (Full View)	1286	5872	2733	4410
% Full view duration	43.61%	48.9%	66.43%	40.09%

Table 3.2: Table showing the ‘value’ of the full court view in terms of broadcast time occupied by this single view.

points in some distributions, relevant data can be extracted by applying thresholding constraints. The first technique involves segmenting the playing area from the background by applying an empirically derived threshold to the differences of colour planes while direct thresholding of the luminance component was performed on the tennis footage to segment the delineating playing area.

Snooker

In snooker, it is noticeable that the playing area is of a clearly contrasting colour to that of the background (figure 3.2). The colour of the cloth used on championship tables exhibits high values of green, while having low blue and red content. Using this knowledge, the table can be segmented by thresholding the difference in colour planes according to equation 3.1.

$$t(i, j) = \{(G(i, j) - R(i, j)) > \tau\} \wedge \{(G(i, j) - B(i, j)) > \tau\} \quad (3.1)$$

Where R, G and B are the red, green and blue colour components of the image, and t is the binary map of the table for pixel locations (i, j) . For snooker footage a threshold of $\tau = 25$ was used to generate the binary image.

The choice of threshold is reflected in the scatter plots of figure 3.1. The difference in colour planes ($G - R$ v $G - B$) for three separate stills of the global view, from each footage source are plotted. The green points correspond to the table area, which was manually extracted. The red points are the remaining colours in view. A contour plot is overlaid to highlight high density regions. The high density region in the top right of the plots, emphasised by the contour lines, is due to the table while the other high density regions are a result of the background surfaces. Note the presence of some isolated green points outside the threshold range ($\tau = 25$). These are attributed to the balls on the table and possibly an encroaching player. Red values within the threshold range are cushion pixels (of similar colour to the table) which were not taken into account when manually extracting the table region for analysis.

Tennis

No matter the type or colour of tennis court surface, the court lines are always painted white. A simple, direct thresholding of the brightness component (V) from HSV colour space, looking for regions which are brighter than a specified threshold, should therefore yield the brightest objects in view. These will generally be due to the lines, players and part of the crowd.

While direct thresholding is sufficiently robust for snooker using non-normalised RGB colour space (since the game is played indoors with relatively uniform lighting conditions) this is not the case for tennis footage which is normally played outdoors. Thresholds often have to be chosen empirically meaning that over a large sample of data, the technique may

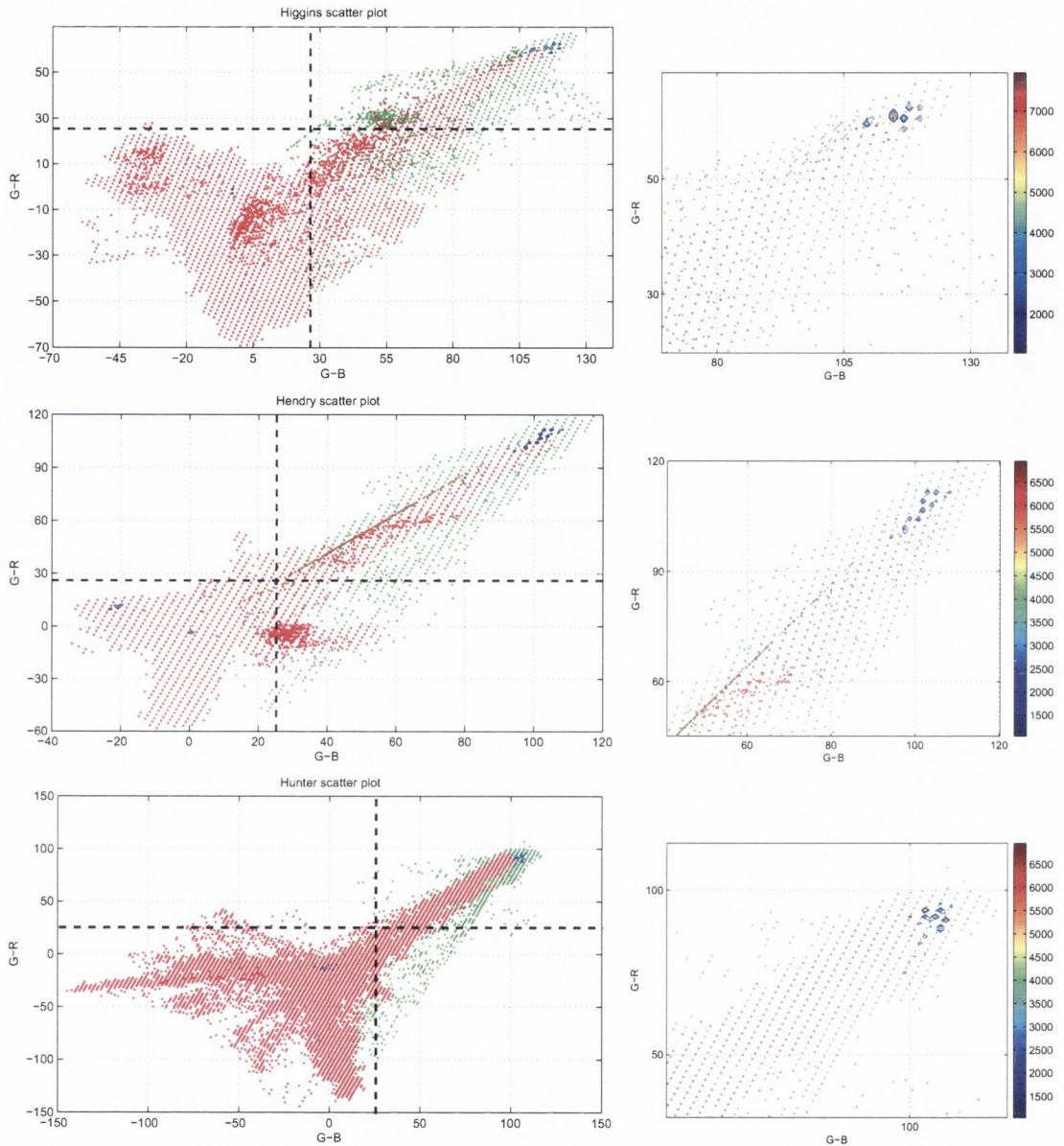


Figure 3.1: A scatter plot of the difference of colour planes for *Higgins* (top), *Hendry* (middle) and *Hunter*. A close-up of the regions of table colour is shown on the right with a colour bar conveying the contour density. The bold dashed line is the threshold values of 25.

not be quite so robust due to possible variations causing drifts in the data. It can be seen in the bottom row of figure 3.3 that the value chosen ($V > 145$) for the brightness threshold does not perform equally well in all sequences. As games are played at different times during the day and in different lighting conditions, court regions can seem brighter and are labelled

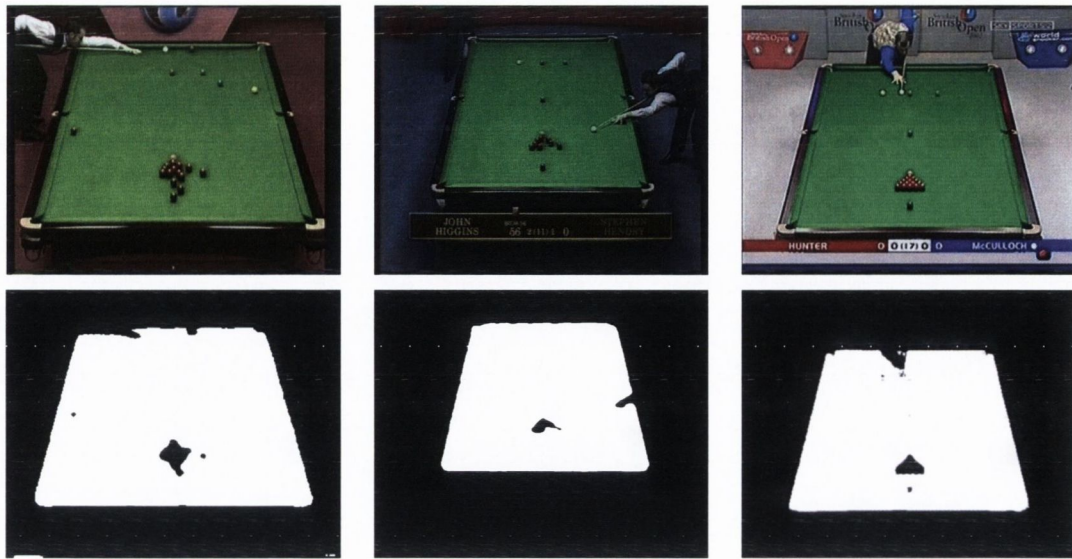


Figure 3.2: Top: Global view from *Higgins*, *Hendry*, *Hunter* sequences. Bottom: Binary maps of the snooker footage generated from RGB colour differences.

as part of the field lines.

3.2.2 Segmentation using adaptive thresholding of colour spaces

In most sports footage, the playing area colours contribute toward a large proportion (typically $\geq 70\%$ in tennis and $\geq 60\%$ in snooker) of the overall colour distribution in the global view. In this view, peaks in the individual distributions correspond to the playing area region (as shown for tennis in figure 3.4). The adaptive thresholding method used for this segmentation uses a greedy algorithm which accounts for this property. The idea is to select $r\%$ of the histogram centred on the mode. The algorithm is outlined in table 3.3 for the brightness component, where the greedy range is $r\%$.

Snooker

Again, *RGB* space is used in this segmentation of the snooker table. Using the same observation as direct thresholding (that there are high values of green for the table in *G* and not in *R* and *B*), the histograms of the difference between the green and blue components, and green and red components are calculated. The binary ‘and’ of the thresholded images $GR_{map} = (G - R)$ and $GB_{map} = (G - B)$, who’s values lie within the 60% greedy range of both histograms are deemed to be the table.

$$table = GR_{map} \wedge GB_{map} \quad (3.2)$$

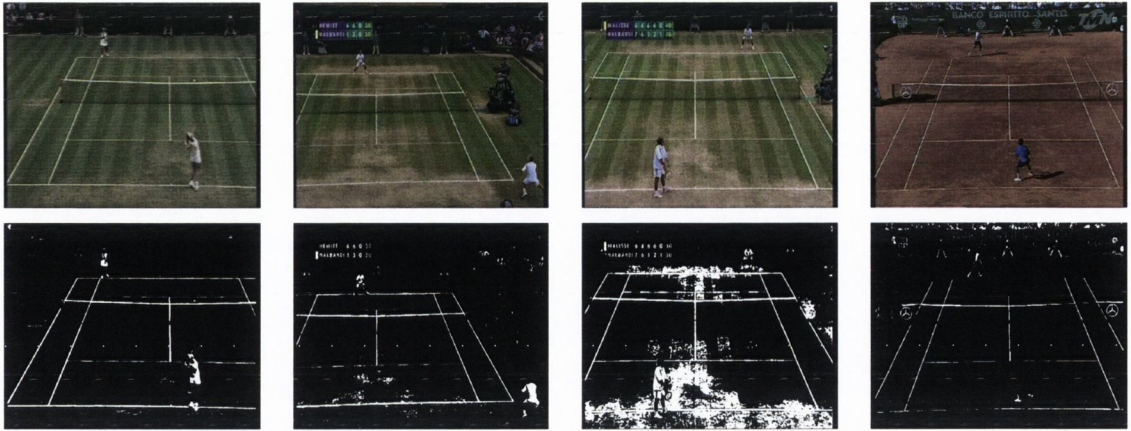


Figure 3.3: Binary maps of tennis footage by application of a direct threshold of $V > 145$. Left to right: *Pierce*, *Hewitt*, *Malisse*, *Costa* sequences.

This shows good results for *Higgins* and *Hunter* footage (figure 3.5). However, since the playing area in the global view in *Hendry* is quite small, the mode in the distribution does not correspond to the colour of the table, but the background. This issue affects the discussion in section 3.2.4.

Tennis

The contrasting luminance and saturation values of the tennis court lines and the court itself offers the possibility of segmenting the court from the delineating field lines. *HSV* (Hue, Saturation, Value) colour space is used for segmenting the tennis court. The peaks in the value, or brightness, and saturation histograms correspond to the dominant colours of the playing area, are firstly found. A greedy algorithm is then used to find the values which account for 65% of the brightness histogram, grown outwards from the peak value and 70% for saturation. As white has a high brightness and low saturation, pixels with values greater than the range produced by applying the algorithm to the brightness component are considered to be non-court surface pixels. Those values less than the range returned by the same algorithm on the saturation component also contribute to the court lines. While the 65% range used for the luminance histogram will uncover some patches of worn down grass, it is necessary to ensure that all the straight lines are detected. The binary map of the ‘and’ operation between the two thresholded colour spaces, V_{map} and S_{map} , retrieves the court lines (equation 3.3).

$$court = V_{map} \wedge S_{map} \quad (3.3)$$

It can be seen in figure 3.3 that the greedy histogram segmentation, with a greedy range

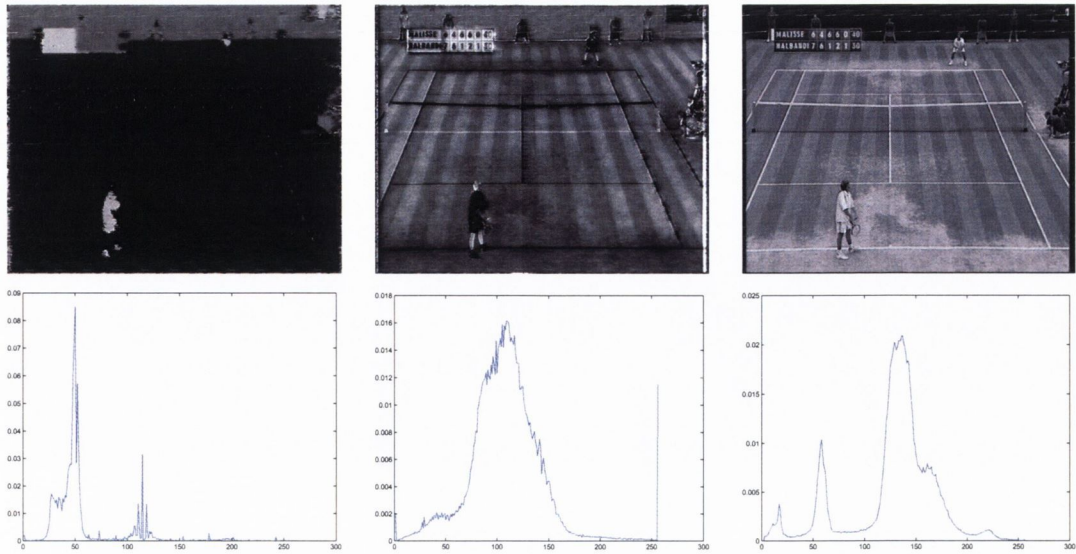


Figure 3.4: Tennis court colour (HSV) distribution. The main lobe in the distribution corresponds to the court surface. (Hue (left), saturation (middle), value (right)).

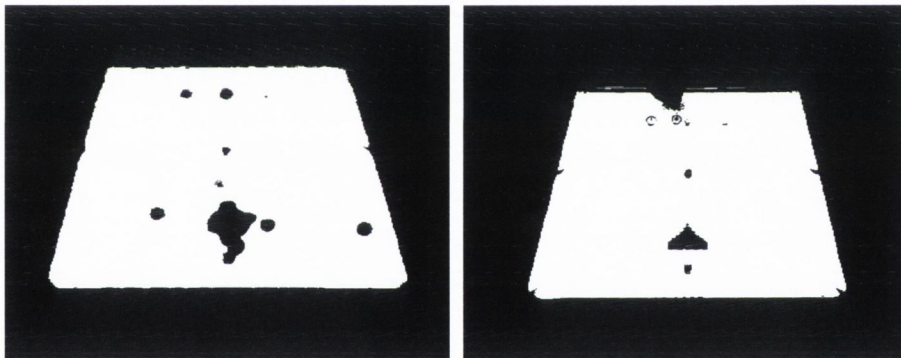


Figure 3.5: Segmentation of the snooker table using the greedy algorithm on Higgins (left) and Hunter (right).

of 65% for the luminance histogram and 70% for the saturation histogram, works well on two types of tennis court surface (grass and clay).

3.2.3 Colour space modelling

In an attempt to derive a more generic solution to the segmentation problem, a probabilistic approach was considered by modelling the colour distribution using a mixture of Gaussians.

1. Compute the normalised M bin brightness (V) histogram, $H_V(m)$ from HSV space.
2. Find the argument of the mode in the distribution.

$$\hat{m} = \arg \max_{1 \leq m \leq M} (H_V(m))$$
Define the variables $k_1 = k_2 = \hat{m}$ for the first iteration.
3. while $(\sum_{n=k_1}^{k_2} H_V(n)) \leq r\%$,
if $(\sum_{h=k_1}^{k_2+1} H_V(h) \geq \sum_{g=k_1-1}^{k_2} H_V(g))$,

$$k_2 = k_2 + 1$$
else,

$$k_1 = k_1 - 1$$
end;
end;

Table 3.3: Greedy histogram algorithm for calculating $r\%$ of the brightness histogram V , from HSV colour space.

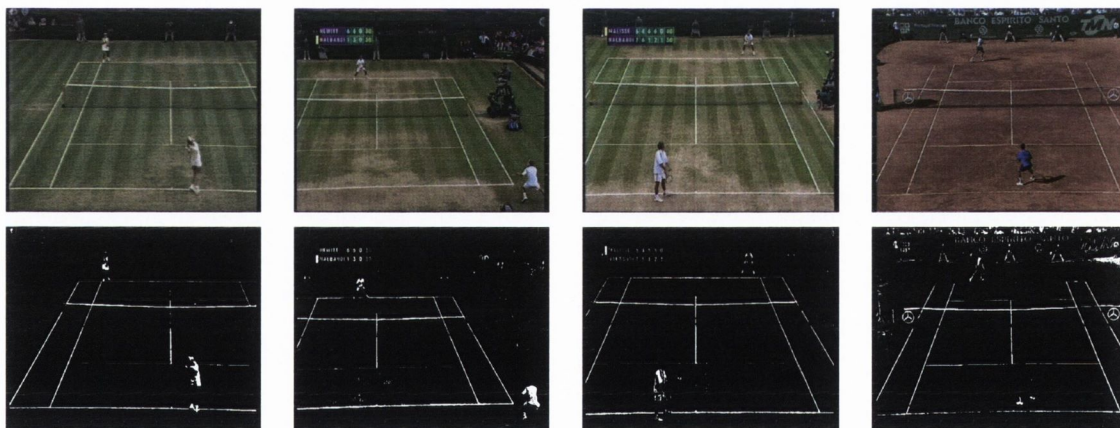


Figure 3.6: Binary maps of tennis footage generated using the greedy histogram. Left to right: Pierce, Hewitt, Malisse, Costa sequences.

In order to characterise the colour content of the footage, the CIE $L^*a^*b^*$ colour space was used.

Depending on the environment (indoor/outdoor) in which a sport is being played, lighting conditions will affect the segmentation process, both temporally and spatially. Under conditions where artificial lighting is present, the brightness over the playing area may not be sufficiently uniform to allow segmentation using the luminance component. The luminance invariant chromaticity space a^*b^* is therefore employed.

In an outdoor environment, the opposite is the case ³. The brightness may vary temporally over the duration of a game. However, the duration in which it takes for lighting conditions to change are considerably longer relative to the frame operations. Moreover, in an outdoor environment luminance should be spatially uniform, assuming there is no shadowing of the playing area due to obstructing objects. A model which incorporates luminance must therefore be considered. An $L^*a^*b^*$ colour model is therefore adopted for outdoor sports.

Multimodal colour space modelling

As a result of the nature of the global view, the playing area accounts for the majority of the total colour distribution over the entire image. The distribution will in general be multimodal but will exhibit a substantial peak in the colour histogram due to the dominant colour of the playing surface, as discussed previously. In order to derive a model for the colour distribution, means and covariances must be resolved from the data. A model is trained by manually selecting 3 regions of playing area from 20 frames at different points in each of the footage sources. A parametric model is then created by approximating the complex pdf in the form of an aggregation of individual Gaussian components (*i.e.* a Gaussian mixture model (GMM)) [13, 37]. Essentially, the goal of the GMM algorithm is to estimate the means, covariances, and probabilities of each mixture distribution. The GMM is also used in chapter 6 for clustering and quantisation of a two dimensional space. A description of the GMM and the iterative estimation formulae are now presented.

Gaussian mixture modelling

In this type of clustering each cluster is mathematically represented by a Gaussian distribution. The entire data set, $\{\underline{x}_n\}_{n=1}^N$, (where \underline{x}_n is a collection of features) can be modelled by a weighted mixture of multivariate Gaussians, each with a particular mean value, μ_k , and covariance matrix \mathbf{R}_k . The Integrated Completed Likelihood (ICL) [11] is known to help in determining the number of mixture components to fit the data. The ICL however was not used in this work. The expectation-maximisation (EM) algorithm is used to iteratively update the parameters of each mixture until some convergence criterion is reached. The mixture is defined as:

$$\mathcal{N}_k \triangleq \mathcal{N}(w_k, \underline{\mu}_k, \mathbf{R}_k) \quad (3.4)$$

³In this thesis, it is assumed that outdoor sports are only considered to take place during the daytime.

For a Gaussian mixture model, the likelihood is given by equation 3.5 where $\Theta = \{w_k, \underline{\mu}_k, \mathbf{R}_k\}$ and C_k is the cluster class

$$p(\underline{x}_n | C_k, \Theta) = \frac{1}{\sqrt{2\pi|\mathbf{R}_k|}} e^{-(\underline{x}_n - \underline{\mu}_k)\mathbf{R}_k^{-1}(\underline{x}_n - \underline{\mu}_k)^T} \quad (3.5)$$

A new parameter set is estimated by maximising Θ to generate a new updated Θ' according to equation 3.6⁴.

$$\Theta' = \arg \max_{\Theta} \left[P \left(\{\underline{x}_n\}_{n=1, \dots, N} | \Theta \right) \right] \quad (3.6)$$

The optimisation process is described below where K Gaussians are fitted to the data \underline{x}_n , where $n = 1, \dots, N$.

1. Initialisation: Randomly choose K points as the centroids of the Gaussians from the data set. The weights on each Gaussian, or mixing coefficients, w_k , are initialised as $w_k = 1/K$. Variances are initialised to be one with zero covariance.
2. Update: *E-Step*: Compute the probability given in equation 3.7. This is known as the mixture ‘responsibility’, so named as it effectively measures how responsible the k^{th} mixture is for generating the data \underline{x}_n .

$$p(C_k | \underline{x}_n) = \frac{p(\underline{x}_n | C_k)p(C_k)}{\sum_{n=1}^N p(\underline{x}_n | C_k)p(C_k)} \quad (3.7)$$

3. Update: *M-Step*: New parameters are estimated using the update equations below based on the mixture responsibilities from the E-step and the data.

$$\begin{aligned} \hat{\underline{\mu}}_k &= \frac{\sum_{n=1}^N p(C_k | \underline{x}_n) \underline{x}_n}{\sum_{n=1}^N p(C_k | \underline{x}_n)} \\ \hat{\mathbf{R}}_k &= \frac{\sum_{n=1}^N p(C_k | \underline{x}_n) (\underline{x}_n - \hat{\underline{\mu}}_k)(\underline{x}_n - \hat{\underline{\mu}}_k)^T}{\sum_{n=1}^N p(C_k | \underline{x}_n)} \\ \hat{w}_k &= \frac{1}{N} \sum_{n=1}^N p(C_k | \underline{x}_n) \end{aligned} \quad (3.8)$$

4. Termination: The algorithm converges if the change in an error function (given by the ratio of likelihoods from the current iteration and previous iteration in equation 3.9) is less than a specified tolerance.

$$\Delta^{(i)} = - \ln \frac{p^{(i)}(\underline{x}_n)}{p^{(i-1)}(\underline{x}_n)} \quad (3.9)$$

⁴The parameter update equations are derived by introducing an auxiliary function $Q(\Theta, \Theta')$ which is the expected value of the complete data log-likelihood function. This is described at length in Bilmes [13].

Where,

$$p^i(\underline{x}_n) = \sum_{k=1}^N p(\underline{x}_n|C_k)p(C_k) \quad (3.10)$$

If $\Delta^{(i+1)} < tol$ terminate, otherwise go to step 2.

Having established a parametric model for the colour distribution ($\Theta = \{\hat{w}_k, \hat{\underline{\mu}}_k, \hat{\mathbf{R}}_k\}$), the likelihood of each pixel, $\{\underline{x}_n\}_{n=1}^N$, is computed and summed over all mixtures K .

$$p(\underline{x}_n|\Theta) = \sum_{k \in K} w_k e^{-0.5(\underline{x}_n - \hat{\underline{\mu}}_k)^T \hat{\mathbf{R}}_k^{-1} (\underline{x}_n - \hat{\underline{\mu}}_k)} \quad (3.11)$$

Tests were carried out on two examples of indoor sports (snooker and badminton) and two outdoor sports (tennis and cricket).

Indoor sports

The a^*b^* chromaticity space results in a good segmentation of the snooker table. Three regions, of approximately 200×200 pixels, from the three source footages listed in section 3.2 are used to train the snooker model. Figure 3.7 shows a contour plot of the 2D a^*b^* histogram with an overlay of the modelling Gaussians for snooker and badminton footage.

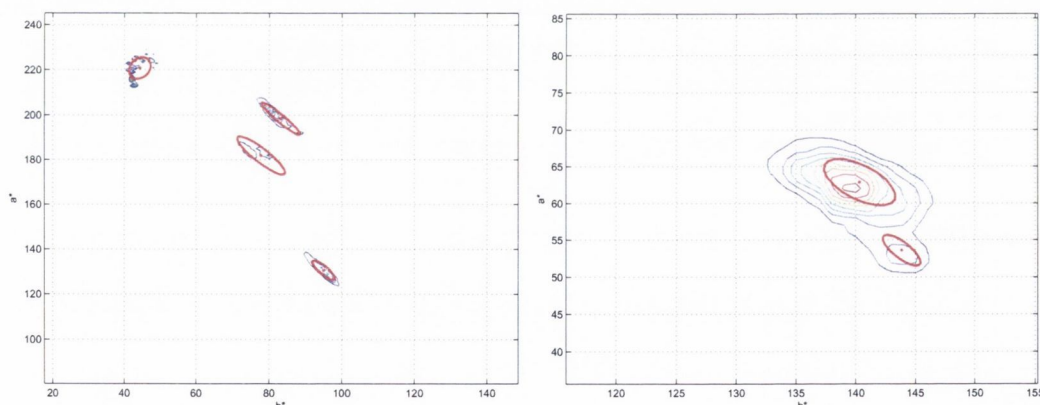


Figure 3.7: Indoor models: Left to right. The a^*b^* distribution for snooker footage approximated using a 4 mixture GMM; The a^*b^* distribution for badminton footage approximated using a 2 mixture GMM.

Outdoor sports

As with the indoor footage, three regions of approximately 200×200 pixels are chosen manually from the source footage to model the dominant colour of the playing area. The scatter

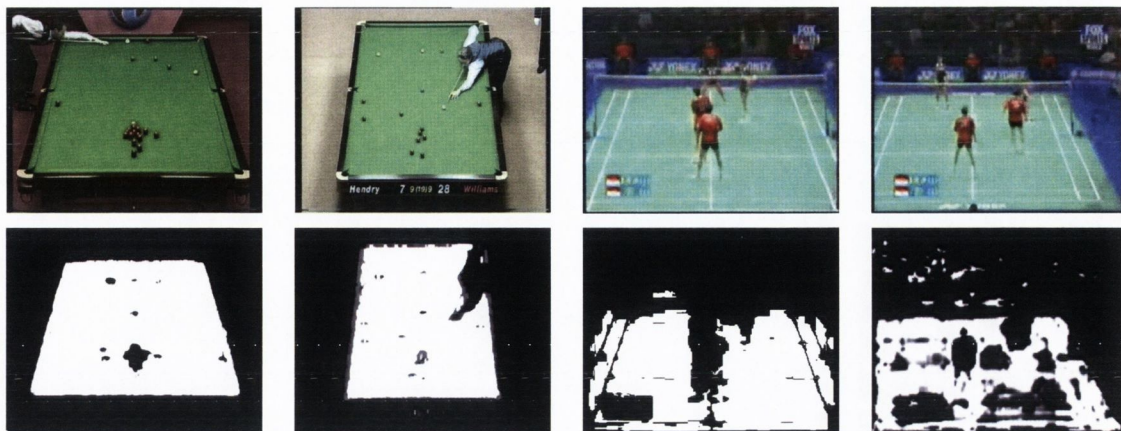


Figure 3.8: Binary maps of snooker and badminton generated using the ab colour model.

plot of the tennis footage exhibits three distinct clusters, while two clusters are evident in the cricket footage (figure 3.9). The regions chosen manually to model the tennis playing area colour were selected to include worn regions of the court resulting in the second peak in the distribution⁵. Three regions of the crease were also chosen from cricket footage. Figure 3.9 shows the 3D $L^*a^*b^*$ scatter plot with an overlay of the three modelling Gaussians for tennis and the two Gaussians required to model the colour distribution of the crease for the cricket footage.

Distinguishing between the tennis court playing surfaces and thereby choosing the correct model type for segmentation can be done by analysing the (r, g) chrominance content of the full court view. As a clay court contains higher red values than grass it is reasonable to say that if equation 3.12 is true, the clay court model should be chosen.

$$\frac{\sum_{i=1}^N \sum_{j=1}^M r(i, j)}{NM} > \frac{\sum_{i=1}^N \sum_{j=1}^M g(i, j)}{NM} \quad (3.12)$$

Where M, N are the number of rows and columns in the image. The opposite is the case then for the grass model. An illustration of the segmentation achieved using the GMM is shown in figure 3.10.

3.2.4 Choice of segmentation method

In summary, three segmentation approaches were considered, direct thresholding, adaptive thresholding and colour distribution modelling with the aim of segmenting the delineating playing surface in snooker and tennis. Direct thresholding was chosen for snooker segmenta-

⁵It was necessary to model the worn regions, as grass tennis courts tend to become degraded following considerable play.

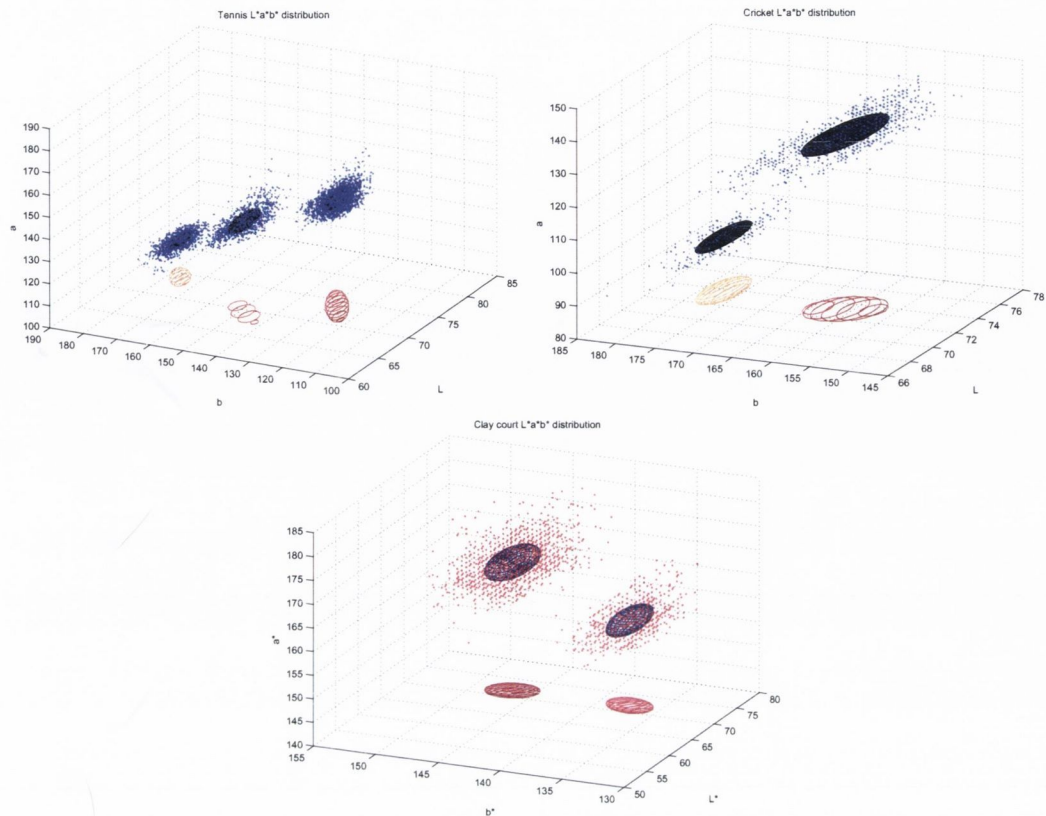


Figure 3.9: *Outdoor models. Clockwise from top left: The $L^*a^*b^*$ plot for grass court tennis footage approximated using a 3 mixture GMM; The $L^*a^*b^*$ plot for cricket footage approximated using a 2 mixture GMM; The $L^*a^*b^*$ plot for clay court tennis footage approximated using a 2 mixture GMM; The data sets for all three footage sources have been subsampled for viewing purposes.*

tion and adaptive thresholding for tennis. The reasons for which are given below.

Snooker

Playing conditions in snooker tend to be stable since the game is played indoors, without any natural light and on a surface which does not vary from competition to competition. Furthermore, the lights are set up so as not to cast shadows on the playing surface. This means that a direct thresholding approach is sensible for segmenting the snooker playing surface from all footages. Of the other two segmentation methods, adaptive thresholding cannot be guaranteed to work on all snooker footage sources since it is assumed that the table accounts for the majority of the view, which it sometimes does not, while the computational

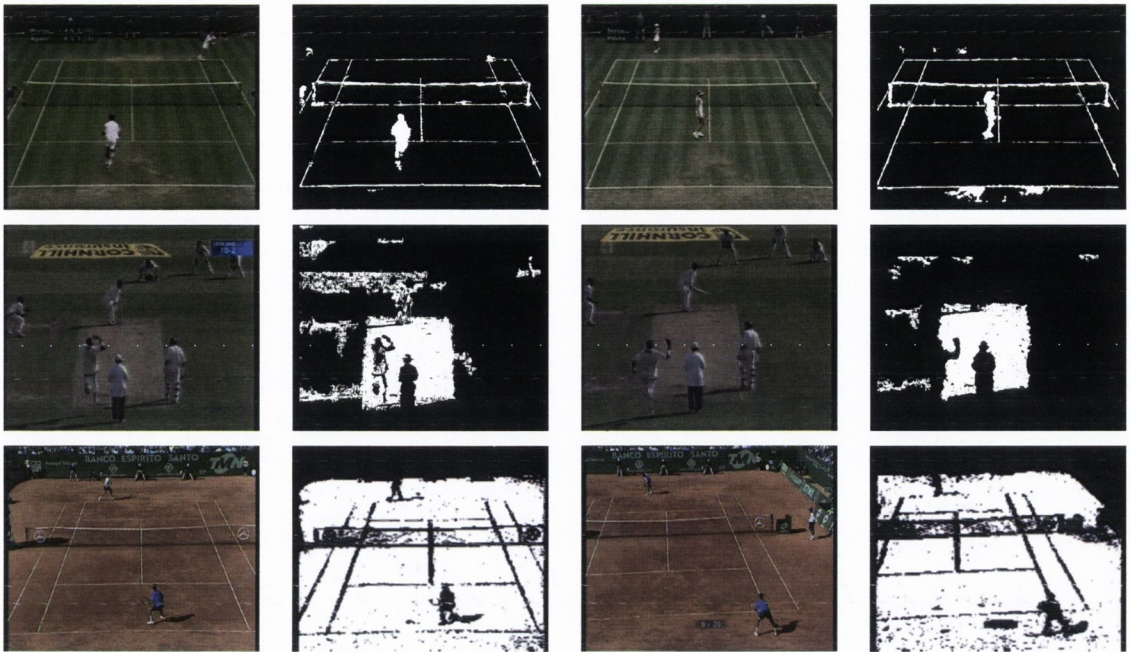


Figure 3.10: Binary maps of grass court tennis (top - the negative is shown here), cricket (middle), and clay court tennis (bottom) footage generated using the $L^*a^*b^*$ colour model.

burden of the GMM is excessive.

Tennis

It is clear from figures 3.3, 3.6 and 3.10 that the GMM and adaptive thresholding offer the most reliable segmentation. While direct thresholding does work well for some sources, the variations in lighting conditions limit its effectiveness. Once again the GMM is not chosen due to its excessive computation, so the adaptive threshold is used.

3.3 Playing area detection and inference of geometry

In the subsequent sections, the results from the segmentation are drawn upon to infer the geometry of the playing areas for tennis and snooker. The Hough transform and the related Radon transform [36] have been used for detecting objects that can be specified by some parametric form (circles, lines, ellipse). The discrete form of the Radon transform, the Mojette transform [35], has also been used for object finding purposes. These transforms have been exploited in a number of applications including medical imaging [51, 127] and cartography [149]. They work by mapping lines in image space to points in Radon or Hough

space by re-parametrisation. A brief review of the Radon transform is given in Appendix D.

To our best knowledge, the work presented in this thesis is the first to exploit the Radon transform for detection of delineating field lines in broadcast sports footage [39]. Both the Radon transform and Hough transform have found considerable success in the sports domain [69, 77, 99] since, due to their robustness against occlusion and relative simplicity. The techniques involved in retrieving the relevant lines and salient points on tennis and snooker playing surfaces are discussed in the subsequent sections.

3.3.1 Geometry of a snooker table

An edge map of the binary image generated by the segmentation in section 3.2 is created using a Sobel Edge detector [71]. A Radon transform is performed on the edge image using polar line parameters (ρ, θ) . Figure 3.11 shows examples of the Radon transform from the segmented snooker table.

From analysis of the footage (and as can be seen in figures 3.11), the orientation of the table in the global view dictates that straight lines should be found at angles in the range $[3^\circ, \dots, 25^\circ]$, $[89^\circ, \dots, 90^\circ]$, and $[155^\circ, \dots, 177^\circ]$. Computational complexity of the Radon transform is substantially reduced by making use of this prior information. Figure 3.11 shows the arrangement of the peaks in Radon space for a full table view. So for a full table view the following pattern is observed:

- One peak in the range $\theta \in [3^\circ, \dots, 25^\circ]$ representing the line at the right hand side of the table.
- Two peaks in the range $\theta \in [89^\circ, \dots, 91^\circ]$ representing the two horizontal lines at the top ($\rho > 0$) and bottom ($\rho < 0$) of the table.
- One peak in the range $\theta \in [155^\circ, \dots, 177^\circ]$ representing the line at the left hand side of the table.

If peaks in Radon space are found in this configuration a table is deemed to have been found.

By finding the intersection points of the retrieved lines from Radon space, the corner pockets can be recovered using equation 3.13, where the equation of a line in polar form is $\rho = x \cos\theta + y \sin\theta$. All relevant global view clips can be extracted using this process.

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos \theta_1 & \sin \theta_1 \\ \cos \theta_2 & \sin \theta_2 \end{bmatrix}^{-1} \begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix} \quad (3.13)$$

Knowing that the diagonals of a trapezoid intersect at its centre enables the coloured ball-spot positions to be recovered. All spots along the centre line of the table can be related to the appropriate subdivision of the table using this process. For example, finding the intersect of the two main diagonals (top left pocket to bottom right and top right pocket to bottom left) allows the blue spot to be recovered, subdividing again from the middle pockets to

the bottom corners of the table, in the same fashion, allows the pink spot to be recovered, *etc.* Furthermore, from the known physical geometry of the table, the yellow and green spots (which are to the 14.5 cm to the left and right of the brown ball) can also be recovered. Results from the table, pocket and spot finding procedure are illustrated in figure 3.11. Figure 3.12 illustrates non-global views correctly rejected using the playing area detection algorithm.

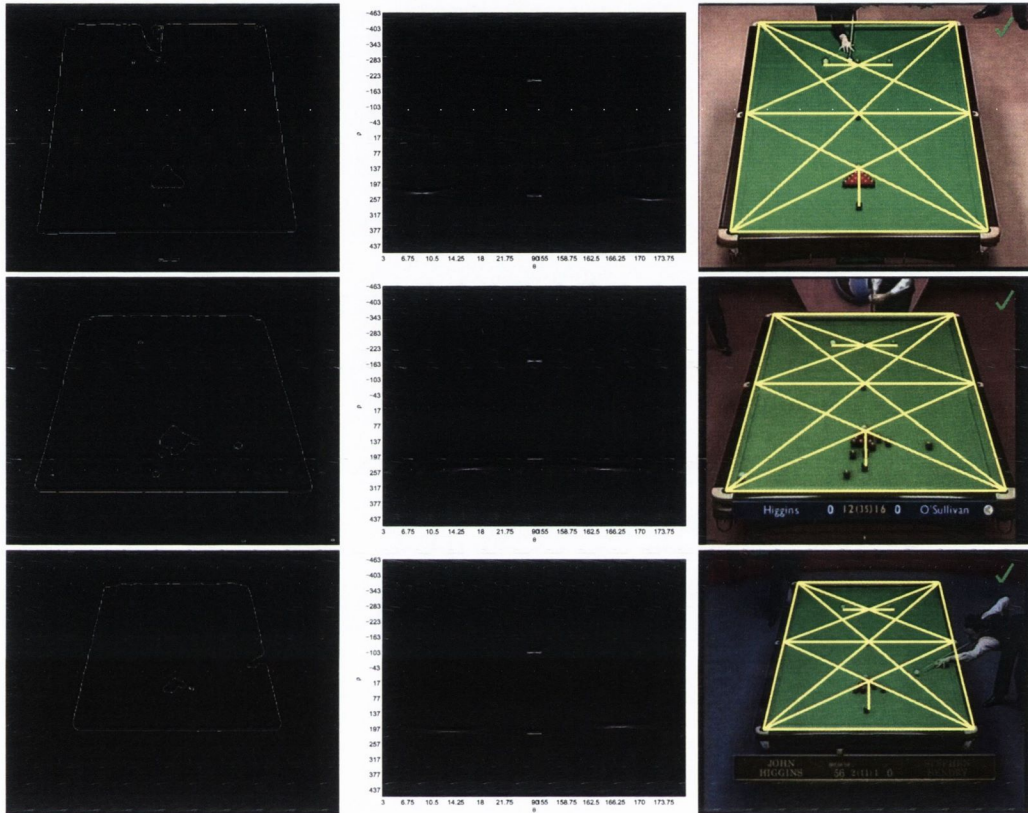


Figure 3.11: Inference of the table geometry for 3 footage sources. Left column: Table edge images; Middle column: Radon Transform of the global view; Right column: Table geometry, spots and pockets recovered.

Experiments for the table finding algorithm were conducted on three sequence. The accuracy of the algorithm is given in terms of precision and recall defined in equation 3.14 where the classification of the retrieved view is given in table 3.4.

$$Recall = \frac{A}{A+C} \quad Precision = \frac{A}{A+B} \quad (3.14)$$

Here, correct views are considered to be those showing the full table and incorrect views are of any other type. Results of the classification showing 100% retrieval for all snooker

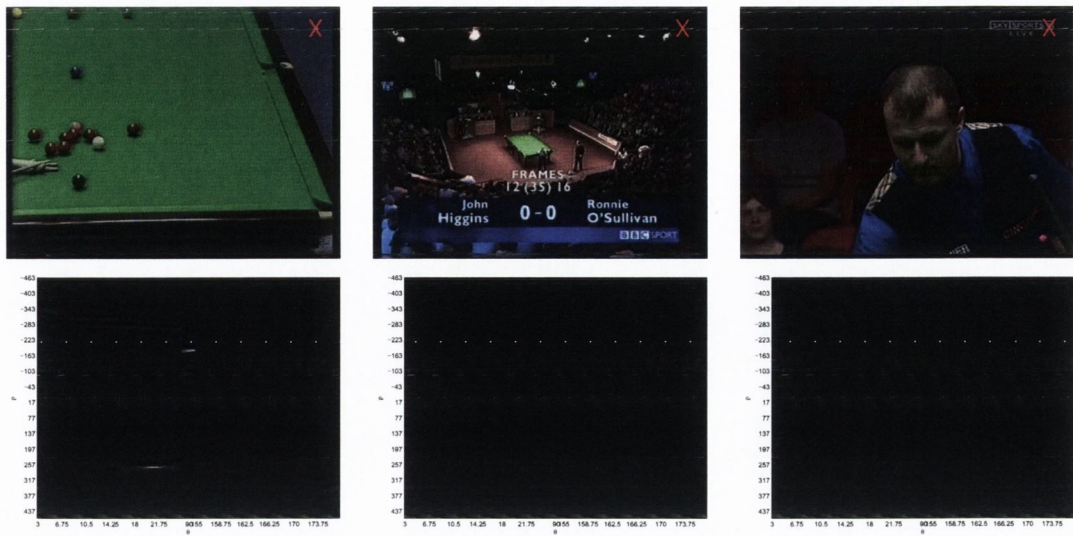


Figure 3.12: Correctly rejected camera views from the snooker footage. The Radon spaces in the second and third images are sparse because no geometry is detected.

	User evaluation	
	Relevant	Not Relevant
Retrieved	A: Correctly retrieved	B: Incorrectly retrieved
Not retrieved	C: Missed	D: Correctly rejected

Table 3.4: Classification of retrieved views.

footage sources are tabulated in table 3.5.

Footage	<i>Hunter</i>	<i>Higgins</i>	<i>Hendry</i>
Precision	100%	100%	100%
Recall	100%	100%	100%

Table 3.5: Precision and recall results for table view classification.

3.3.2 Geometry of a tennis court

Due to the dynamic nature of the game, tennis footage exhibits a great deal of horizontal translational camera motion as the camera pans to follow the main action on court. As a

consequence of the panning, horizontal camera translation in image space results in a vertical translation of the projections in Radon space. This is due to the changes in the parameter ρ (the perpendicular distance from the centre of the image to a line) as the lines drift from their original position. This is shown in figure 3.13, which illustrates a simulated camera translation to the right over two frames and the affect on the Radon transform.

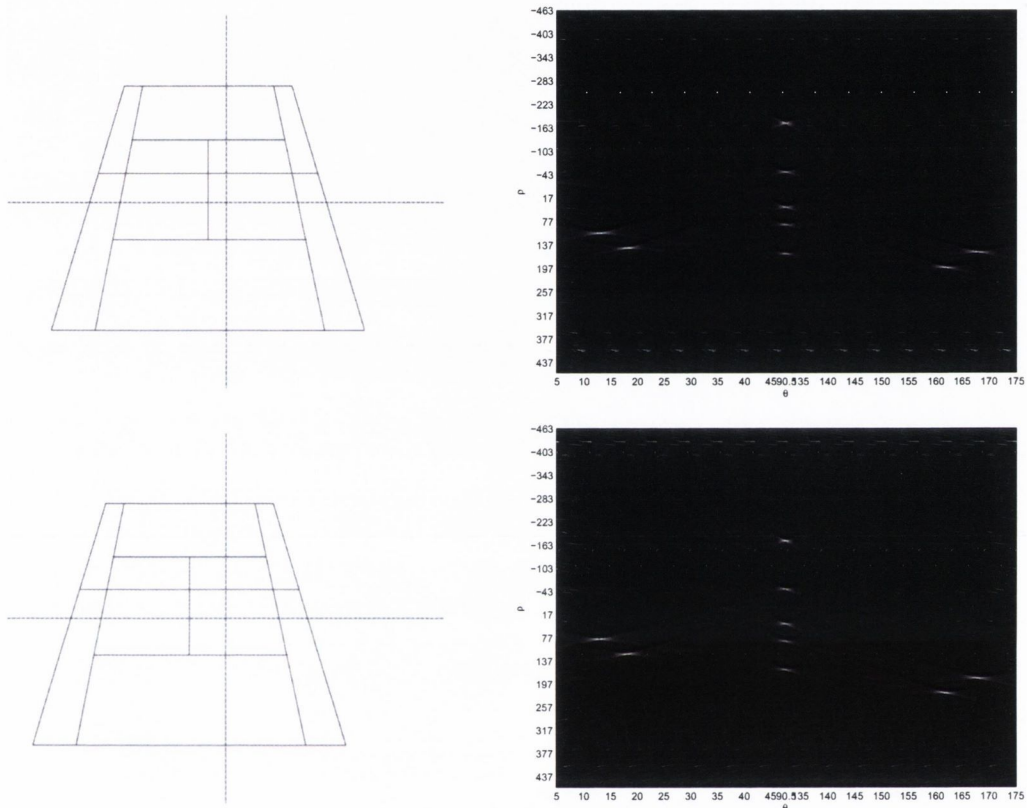


Figure 3.13: Illustrating the resulting Radon transform of a simulated camera pan to the right. Top: Tennis court at frame t and its corresponding Radon transform; Bottom: Frame $t + 1$ and its corresponding Radon transform. The vertical displacement of the transform of the tram lines is clearly visible as the values for ρ change.

Furthermore, since the camera capturing the global view is fixed at the centre of the court, camera panning will cause a perceived rotation of the lines about the fixed location of the camera filming the action. In the case of translation to the left, the lines will appear to rotate clockwise, and counter clockwise for a translation to the right. Consequently, peaks in Radon space will drift horizontally (left or right depending on the rotation of the line) due to the varying line parameter θ . A simulated rotation of the tennis court, and its corresponding

Radon transform is shown in figure 3.14. The perceived rotation of the court from some of the televised footage is well illustrated in figure 3.15.

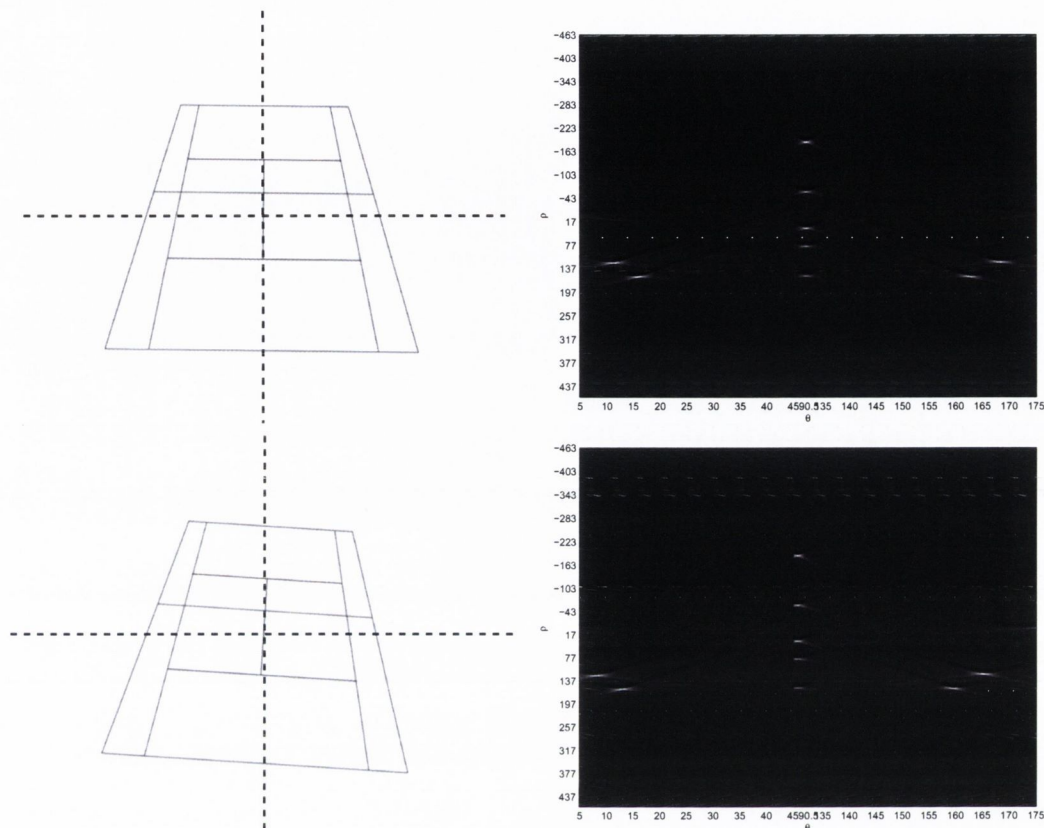


Figure 3.14: *Illustrating the resulting Radon transform of a simulated rotation. Top: Tennis court at frame t and its corresponding Radon transform; Bottom: Frame $t + 1$ and its corresponding Radon transform. The horizontal displacement of the tram lines can be seen as the values for θ change.*

The court can be retrieved without the need for compensating for camera motion. This is achieved by noting that the horizontal lines are always present in the global view as the camera will rarely pan sufficiently causing them to disappear. Furthermore, from analysis of broadcast footage, it was realised that at least 2 vertical tram lines will also be in sight.

Significant peaks in the region $\theta = [5^\circ, \dots, 45^\circ]$ and $\theta = [135^\circ, \dots, 175^\circ]$ in Radon space, represent the two tram lines on either side of the court. This range of angles is sufficient to allow for the effects of camera panning in image space.

The peaks are located by thresholding the individual histograms of the specified θ ranges in Radon space. The minimum of the top 0.25% of the histogram in the range $\theta = [5^\circ, \dots, 45^\circ]$



Figure 3.15: Examples of the perceived rotation of the tennis court.

locates the vertical trams on the right hand side of the court, the minimum of the top 1% of the histogram in the range $\theta = [85^\circ, \dots, 95^\circ]$ locates the horizontal lines and the minimum of the top 0.25% of the histogram in the range $\theta = [135^\circ, \dots, 175^\circ]$ locates the left hand side tram lines. A further condition that the threshold be greater than 50 is also imposed to detect the peaks. The percentages values used to arrive at the threshold value were derived empirically.

The structure of the lines, labelled with distances and angles, and the corresponding Radon transform is illustrated in the mock schematic of a court shown in figure 3.16. The difference in the θ values between the ‘parallel’ tram lines on either side of the court arise from the perspective distortion of the playing area.

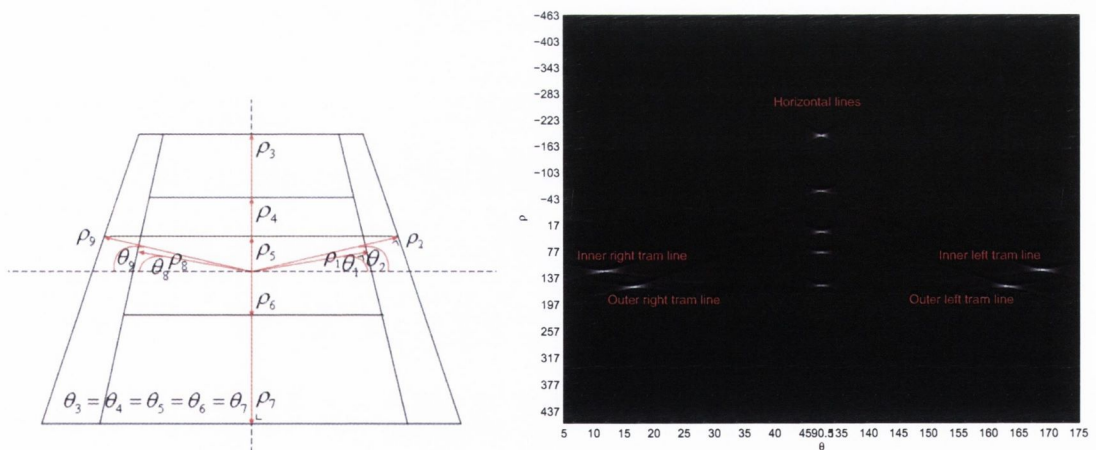


Figure 3.16: Illustration of the tennis court geometry. Left to right: A schematic of a tennis court with parameters (ρ, θ) for each line; The Radon transform of the schematic.

Peaks spanning the range $\theta = [85^\circ, \dots, 95^\circ]$ are deemed to be the horizontals, of which

Vertical $\theta = [5^\circ, \dots, 45^\circ]$	Horizontal $\theta = [85^\circ, \dots, 95^\circ]$	Vertical $\theta = [135^\circ, \dots, 175^\circ]$
2	4 or 5	2
1	4 or 5	1
0	4 or 5	2
2	4 or 5	0

Table 3.6: Conditions on the presence of lines in order for a full table view to be flagged.

there are five. It was found empirically that a range of $\theta \pm 5^\circ$ off the horizontal was sufficient to locate those horizontal lines and compensate for their apparent rotation. Occasionally, the top of the net covers almost all of one of the lines, depending on the angle at which the camera capturing the global view is perched. Either four or five peaks are sought for in this range. Two of the peaks are below the centre of the image (*i.e.* have negative values of ρ) and two or three are above the centre of the image (*i.e.* positive values of ρ). The schematic of the tennis court illustrated in figure 3.16 shows five horizontal lines.

If peaks are not found in the required order shown in table 3.6, or if the total number of peaks is less than 6 (sparse lines) or greater than 10 (spurious lines), then a view other than the tennis court is flagged. Supplemental views can be further categorised using colour and shape features. This is outlined in section 3.3.4.

To eliminate objects which may not be lines, accumulated points in Radon space less than 25 pixels are suppressed. Line intersections are found in the same way as for snooker using equation 3.13.

In order to reconstruct the full shape of the court, tram lines which are out of view as a result of camera translation are simulated by mirroring the existing peaks in Radon space. As discussed previously, horizontal camera translation is manifest as a vertical displacement of the peaks in Radon space. In Radon space, the ρ values of the peaks (corresponding to the horizontal lines) behave asymmetrically about a centre point where the distances to corresponding tram lines are equal. Consequently, this enables the location of “hidden tram lines” (*i.e.* those which are out of view) to be approximated.

To simulate the hidden lines, their (ρ, θ) parameters must be derived from the existing tram line data. In order to achieve this, a frame is extracted from the footage where the difference in distances from the centre of the image of the two outer tram lines is less than 10 pixels (from figure 3.16), where $\rho_1 \approx \rho_8 \approx \rho_0^*$ and $\rho_2 \approx \rho_9 \approx \rho_i^*$ respectively ⁶.

This is referred to as the centre frame. By calculating the drift of each peak in the relevant θ range in the current frame from its location in Radon space in the centre frame,

⁶It is assumed that there is no vertical camera translation so a frame of this type gives the one location where this equality is valid.

the perpendicular distance to the corresponding hidden tram line can be approximated as:

$$\rho_o^{(t)'} = \rho_o^* + (\rho_o^* - \rho_o^{(t)}) \quad (3.15)$$

$$\rho_i^{(t)'} = \rho_i^* + (\rho_i^* - \rho_i^{(t)}) \quad (3.16)$$

Where ρ_o' and ρ_i' are the distances to the hidden outer and inner tram lines respectively and $\rho_o^{(t)}$ and $\rho_i^{(t)}$ are the distances to the outer and inner visible tram lines in the current frame t .

Occasionally, if the camera angle is tight or the segmentation not good, the outside lines are not sufficiently long to be detected using the Radon transform. Under such circumstances, the parameters of those lines have to be inferred. This is simply done by calculating the distance between the inner lines in the centre frame and those in the current frame and offsetting the outer tram lines in the centre frame by the difference in distance.

When calculating the θ parameters of the hidden tram lines, camera induced line rotation must be accounted for. θ values from the centre frame are used for approximating the rotation. Using figure 3.16 as an example of the centre frame, the values for $\theta_1 = \theta_8 = \theta_o^*$ and $\theta_2 = \theta_9 = \theta_i^*$ are registered. In a similar fashion as estimating the values for the ρ parameters, θ values for the right hand tram lines are approximated using equations 3.18 and the left hand tram lines are approximated using equations 3.20.

$$\theta_o^{(t)'} = \theta_o^* + (\theta_o^* - \theta_o^{(t)}) \quad (3.17)$$

$$\theta_i^{(t)'} = \theta_i^* + (\theta_i^* - \theta_i^{(t)}) \quad (3.18)$$

$$\theta_o^{(t)'} = 180 - \theta_o^* + (180 - \theta_o^* - \theta_o^{(t)}) \quad (3.19)$$

$$\theta_i^{(t)'} = 180 - \theta_i^* + (180 - \theta_i^* - \theta_i^{(t)}) \quad (3.20)$$

The resulting angles $\theta_i^{(t)'}$ and $\theta_o^{(t)'}$ are those of the corresponding tram lines on the opposite side of the court, where i stands for inner and o outer. θ_o and θ_i are the angles of the lines in the current frame t . Changes in the angle of the lines as a result of perspective distortion are negligible as the camera pans to the other side of the court, and are not considered in the approximation.

While simulation of lines outside the range of the camera may not be useful for viewing purposes, it does allow reconstruction of the remainder of the court. The same assumption as used for snooker, in that the diagonals of a trapezoid always intersect at its centre is employed to find the centre line. Diagonals from the corners of the outer tram lines (be they simulated or real) intersect at the centre of the court. By consecutive subdivision of the court, the remaining lines can be recovered.

Figure 3.17 shows fully reconstructed tennis courts in the global view. On the left, a court where only the left hand tram lines and horizontal lines are viewable is shown. The middle image illustrates the global view of the tennis court where the inner left hand tram line along with both right hand tram lines is in view. Lastly, a centre frame is shown on

the right. Figure 3.18 illustrates non-global views correctly rejected using the playing area detection algorithm. It is easy to reject these images because the Radon transform does not exhibit the required arrangement of peaks.

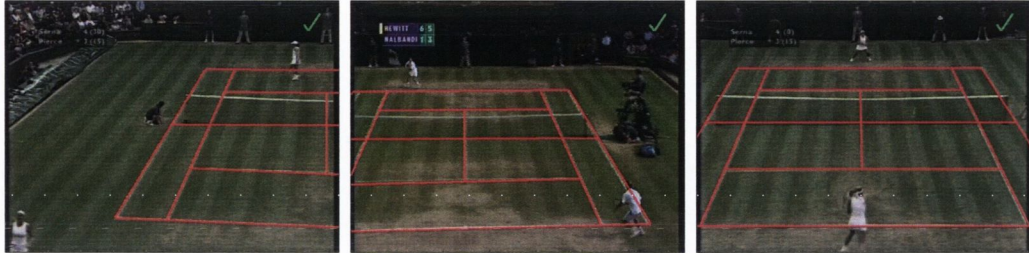


Figure 3.17: Fully reconstructed tennis courts in the global view. The interpolated lines are overlaid on the existing white lines and are shown in red.

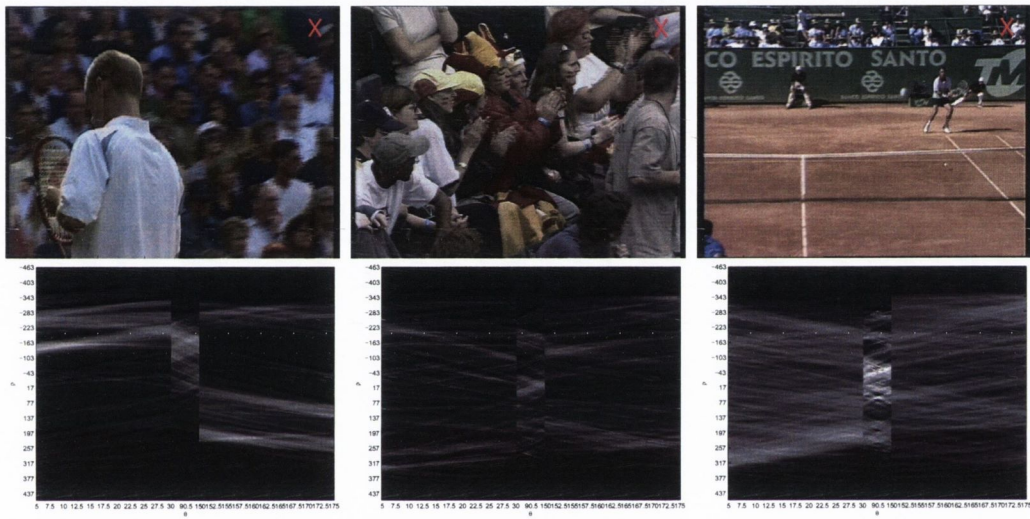


Figure 3.18: Correctly rejected camera views from the tennis footage. The discontinuities on the abscissa are a result of the range of θ used.

Footage	<i>Pierce</i>	<i>Hewitt</i>	<i>Malisse</i>	<i>Costa</i>
Precision	100%	100%	100%	N/A
Recall	100%	100%	99.95%	N/A

Table 3.7: Precision and recall results for tennis court global view detection.

Experiments were conducted on three tennis sequences played on a grass surface and one on a clay court. The lower recall from the *Malisse* footage is a result of poor segmentation, which in turn can be considered to be a consequence of the poor quality of the captured footage. Over a duration of 21 frames, the camera pans very quickly, following the ball from a hard shot. As a result of motion blur the vertical tram lines blend with the court surface reducing their brightness. The segmentation reveals only small parts of the white line and cannot be detected by the court finding algorithm. Results of the tennis court detection are given in table 3.7 in terms of precision and recall.

In an attempt to classify the footage into its further constituent shots, it was parsed according to the statistical moments of local colour and geometrical based features. The problems encountered in classifying the views in the *Costa* footage can be addressed using this method. This is discussed in section 3.3.4.

3.3.3 Radon Moment

Sports such as tennis, snooker, badminton, and cricket all occur within predefined playing limits and are therefore well defined in terms of their geometry. Most of the video footage from these events contains well delineated field lines in the views which contain the most information about the play - for example, the court lines in tennis, and the edge of the table in snooker. It is sensible then that the video should be parsed according to the geometry of the camera view.

Previous work has considered the use of 3D scene geometry [141] to generate a correspondence between certain image features and real court markings. This information could be used also for identifying the camera view, hence allowing the video to be parsed. This can be a complicated exercise, and in fact a much simpler idea yields the same information. What is of interest is the relative geometry of the lines within each image; it is not important to know how that geometry relates to the real world, only how it relates to other geometries, from other views, in the footage.

Summarising the geometry of that edge information in view will yield a useful feature for parsing. The Radon transform of an image containing edge information yields concentrated peaks representing significant straight lines and since shapes of the playing areas for the sports listed are quite distinctive, different views of the playing areas exhibit very dissimilar

arrangements of peaks in Radon space. The nature of this Radon surface will therefore follow changes in the edge information. Summarising the Radon transform should therefore yield an appropriate feature, and it is proposed to use the $p + q$ th order geometric moment [138, 139] as follows:

$$\mu_{pq} = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} (i - x_c)^p (j - y_c)^q f(i, j) \quad (3.21)$$

Where i, j are the pixel co-ordinates, $p + q$ is the moment order and x_c, y_c are the co-ordinates of the origin in Radon space, a single feature describing the frame can be obtained.

Snooker

The different geometries of the snooker table shown from the various camera angles used in common televised footage is reflected in Radon space by exhibiting very distinct transforms. For example, the Radon space of the full table view reveals four distinct peaks representing the edges of the table, while the projection of the table from a different camera angle in Radon space shows a number of peaks in the incorrect order (figure 3.12). While all broadcasters will have preferences about the location of the cameras around the table, the full table view is the most commonly used and can be considered to be of greatest importance as it bears the most useful information.

Figure 3.19 shows the 8th order Radon moment for all the snooker footage. The various plateau level are shown in different colours to highlight the view type. The 8th order moment was chosen empirically, because it gave the best separation in feature space.

Tennis

In section 3.3.2, for all footage sources, the court lines were found to lie at angles in the range $\theta = [5^\circ, \dots, 45^\circ, 85^\circ, \dots, 95^\circ, 135^\circ, \dots, 175^\circ]$. The distances to the visible lines always remains in the range $\rho = [-463, \dots, 463]$, however, the lines which are not in the view can occupy the ranges $\rho = [-1389, \dots, -464]$ and $\rho = [464, \dots, 1389]$ since the position of the tennis court lines change relative to the centre of the image⁷.

Classifying the different views using this method proves more difficult than for the snooker footage since the global view is subject to translational global motion. As the camera pans, both the magnitude of the lines and their positions in Radon space change. The inconsistencies in the Radon moment plots for the tennis footage reflect this and are illustrated in figure 3.20.

⁷These values of ρ assume that the resolution of the images are 720×576 .

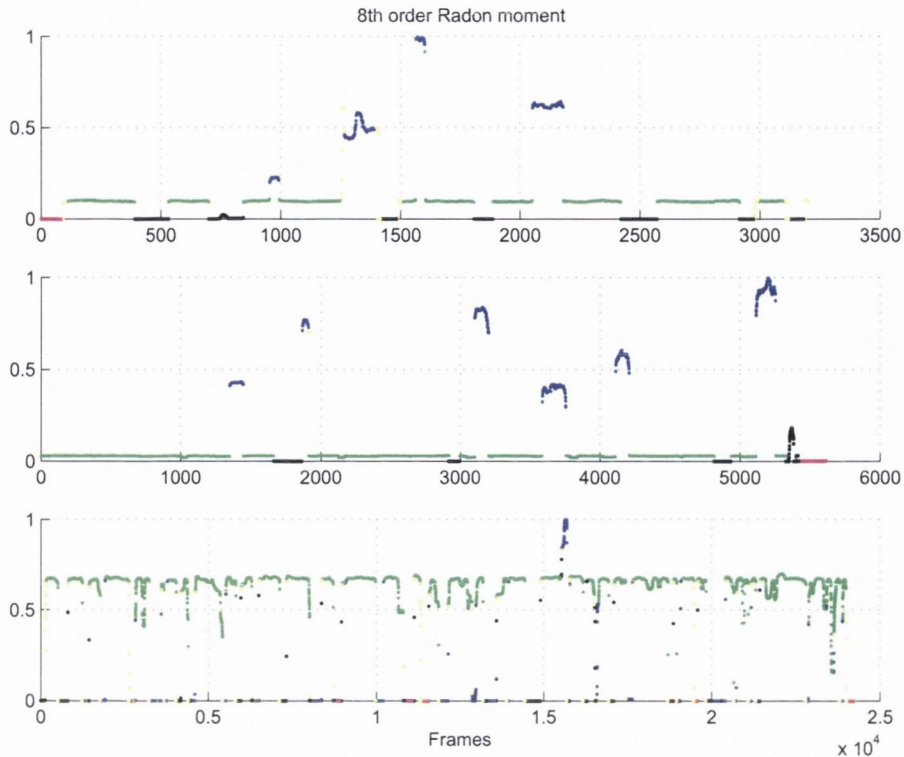


Figure 3.19: 8th order Radon moment for *Higgins* (top), *Hendry* (middle) and *Hunter* (bottom). The green value corresponds to the global view, magenta is a different view of the table, black is a close up of the player, red is the commentator or crowd, blue is a close up of the table and yellow is a gradual shot transition.

3.3.4 Statistical colour and geometrical moments

As discussed in section 3.3.3, the playing areas of both tennis and snooker are well-defined by their geometrical features. Each of the views associated with the different cameras also exhibit differing colour content. Local colour based measures are therefore considered as further indicators to the particular view content [33]. A 3-tuple containing the chrominance and intensity information is defined in equation 3.22 and is a succinct representation of the frame. The red (R) and green (G) colour spaces are normalised by the intensity (equation 3.22) component of the image resulting in the rg chrominance space [122].

$$m^{colour} = \begin{pmatrix} r = \frac{R}{I} \\ g = \frac{G}{I} \\ I = R + G + B \end{pmatrix} \quad (3.22)$$

Further shape features are also considered for classifying the different camera views. These features are not restricted by introducing thresholding constraints and use all the information

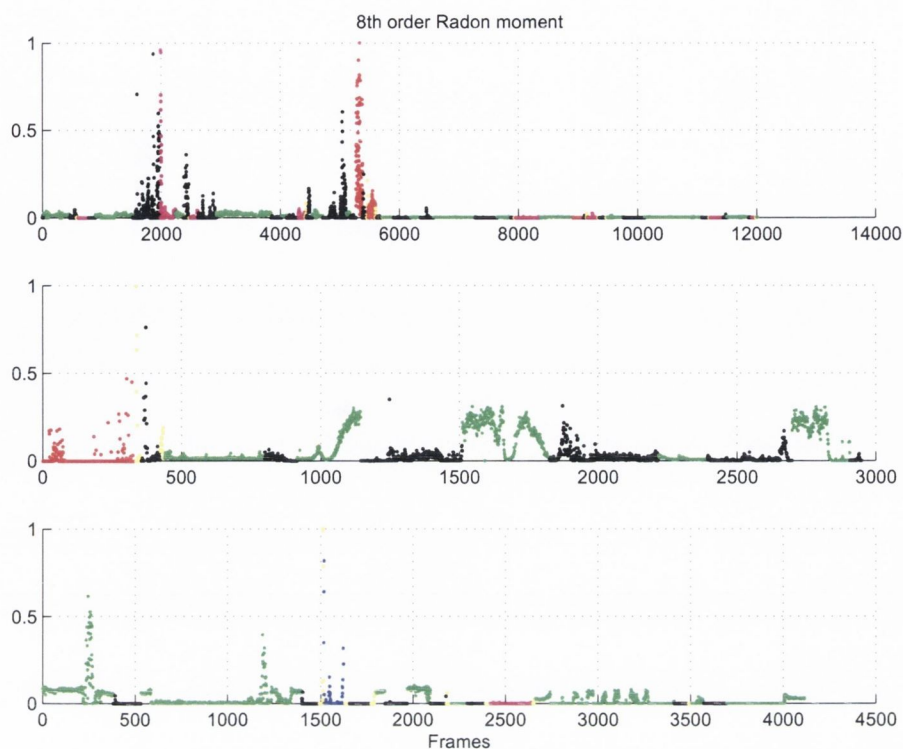


Figure 3.20: 8th order Radon moment for Hewitt (top), Pierce (middle) and Malisse (bottom). The green values corresponds to the global view, magenta is a different view of the court, black is a close up of the player, red is the commentator or crowd, blue is a close up of the court and yellow is a gradual shot transition.

in the image. The first parameter, θ , is the angle of a local edge. The second parameter, α , is an alignment measurement. θ is related to α in that if two points belong to the same straight contour, they will have similar values. The third parameter, N , is the norm of the spatial gradient computed on the intensity component. The 3-tuple containing the shape information is given in equation 3.23. The gradients of the intensity images, $[I_x, I_y]$ are computed using a Deriche operator [30] where the subscript is the gradient direction. A schematic of how the parameters are calculated is shown in figure 3.22. Figure 3.21 illustrates the shape measures using the global views in tennis and snooker footage.

$$m^{shape} = \begin{pmatrix} \alpha = x \frac{I_x}{N} + y \frac{I_y}{N} \\ \theta = \arctan \frac{I_x}{I_y} \\ |N| = \sqrt{I_x^2 + I_y^2} \end{pmatrix} \quad (3.23)$$

By considering the statistical moments of the measures the representation of the features can be reduced to a single value for each image. The extracted features are of very low

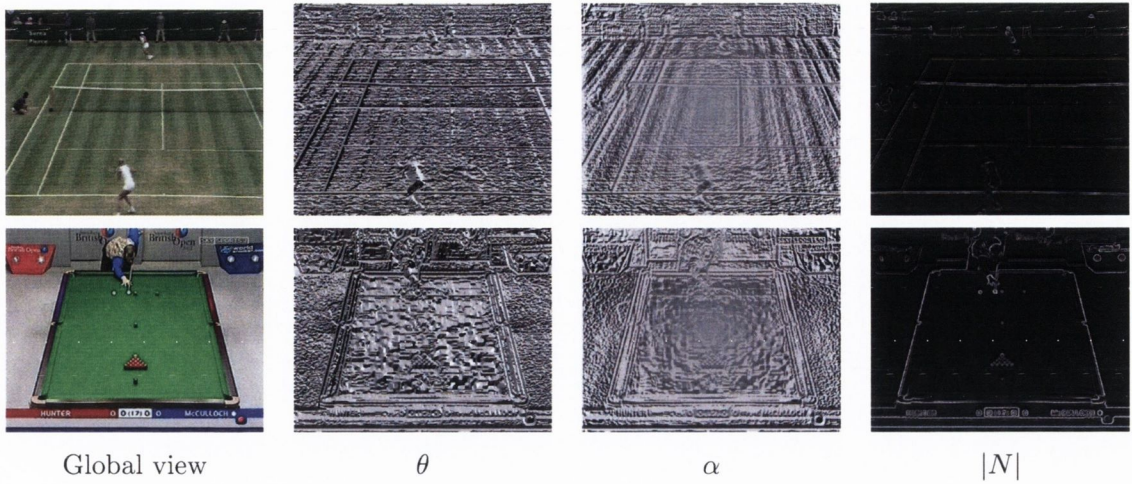


Figure 3.21: Shape features for snooker and tennis footage.

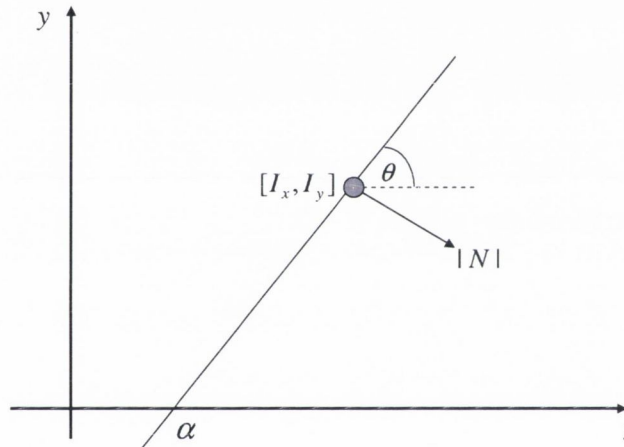


Figure 3.22: Illustration of the geometrical features, θ , α and $|N|$ for a straight line.

computational complexity. The first order moments, which correspond to the mean values of the features are computed on each frame according to equations 3.24 where \mathbf{x} is the spatial location of the local measure and t is the frame number. For moment orders $i + j + k = 1$:

$$M_{ijk}^{colour}(t) = \sum_{\mathbf{x}} r^i(t, \mathbf{x}) g^j(t, \mathbf{x}) I^k(t, \mathbf{x}) \tag{3.24}$$

$$M_{ijk}^{shape}(t) = \sum_{\mathbf{x}} \alpha^i(t, \mathbf{x}) \theta^j(t, \mathbf{x}) N^k(t, \mathbf{x})$$

Higher order statistical moments are calculated by centring the features on their first

order moment value as shown in equation 3.25. So, for moment orders where $i + j + k > 1$:

$$M_{ijk}^{colour}(t) = \sum_{\mathbf{x}} (r(t, \mathbf{x}) - M_{100}^{colour}(t))^i (g(t, \mathbf{x}) - M_{010}^{colour}(t))^j (I(t, \mathbf{x}) - M_{001}^{colour}(t))^k$$

$$M_{ijk}^{shape}(t) = \sum_{\mathbf{x}} (\alpha(t, \mathbf{x}) - M_{100}^{shape}(t))^i (\theta(t, \mathbf{x}) - M_{010}^{shape}(t))^j (N(t, \mathbf{x}) - M_{001}^{shape}(t))^k \quad (3.25)$$

Figures 3.23 and 3.24 show the evolution of the shape and colour moments for the footage *Hendry*. It can be seen that the features occupy different levels for the various view types.

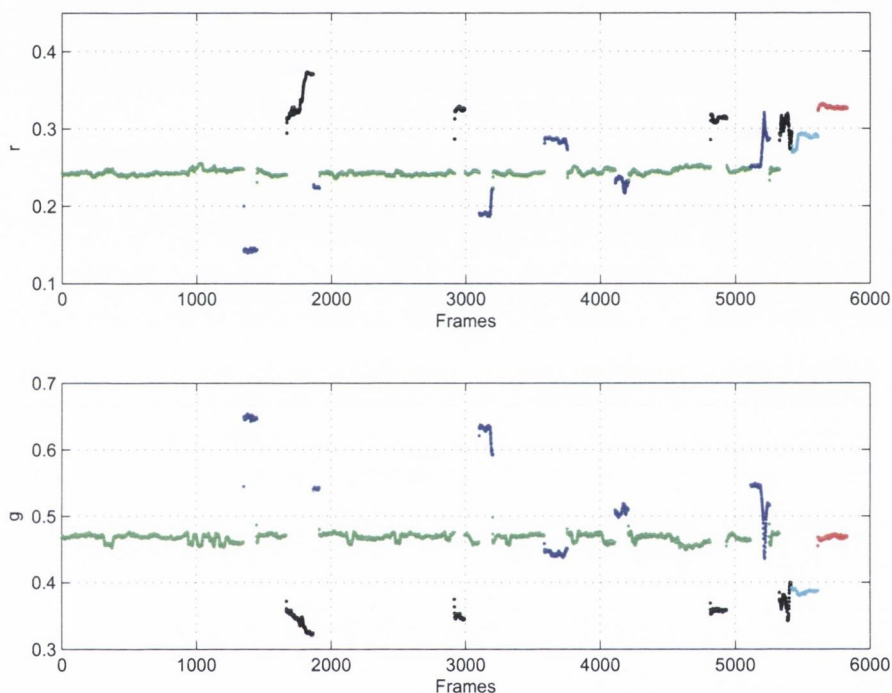


Figure 3.23: Evolution of the mean of r and g for the footage *Hendry*. The green plot is that of the full table, red is the crowd/commentator, blue is a close-up of the table and black is a close up of the player.

Scatter plots of the statistical moments (figure 3.25) show good separation for the different classes of camera view. As can be seen from the plots, the frames of interest exhibit relatively homogeneous moment values. The stochastic nature of the feature will be modelled using a hidden Markov model which will enable the various shots to be classified as a particular view type. This will be presented in chapter 6.

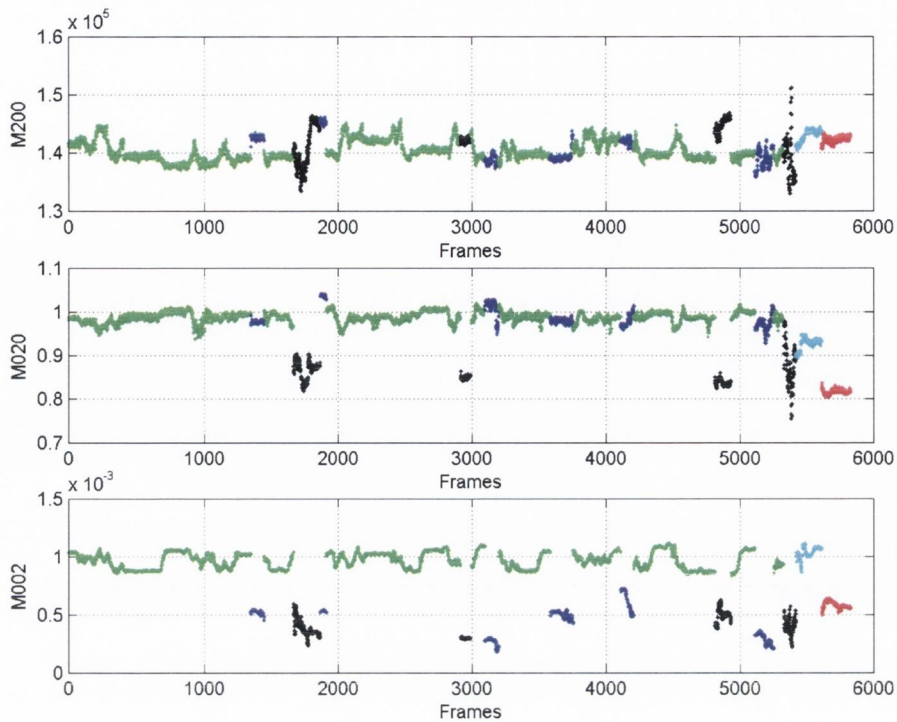


Figure 3.24: *Plots of the second order moments of the statistical shape features for Hendry. From top to bottom: α , θ , N . The colours correspond to those camera views given in figure 3.23.*

3.4 Temporal boundary detection

The temporal unit most commonly used for video analysis is the shot. It is typically punctuated by gradual or sharp transitions or event specific wipes. Sharp transitions are the most easily detected. Their position in the video stream can be located by exploiting the correlation between consecutive frames in terms of their colour, luminance or other local features.

Gradual transitions such as fades and dissolves are more difficult to detect. They result from intensity scaling of frames in a shot. Dissolves are a mixture of two fades where the intensity of one shot is scaled up and the other is reduced. Consequently, the two shots are both spatially and temporally intermingled. Wipes are an editing effect which are broadcaster or event specific. However, they all exhibit the same property in that one shot is gradually spatially replaced by another. As wipes and mattes are used less frequently than the other transitions mentioned [14], only shot cuts and dissolves are sought for. For the sports footage used in this thesis, shot transitions are detected using:

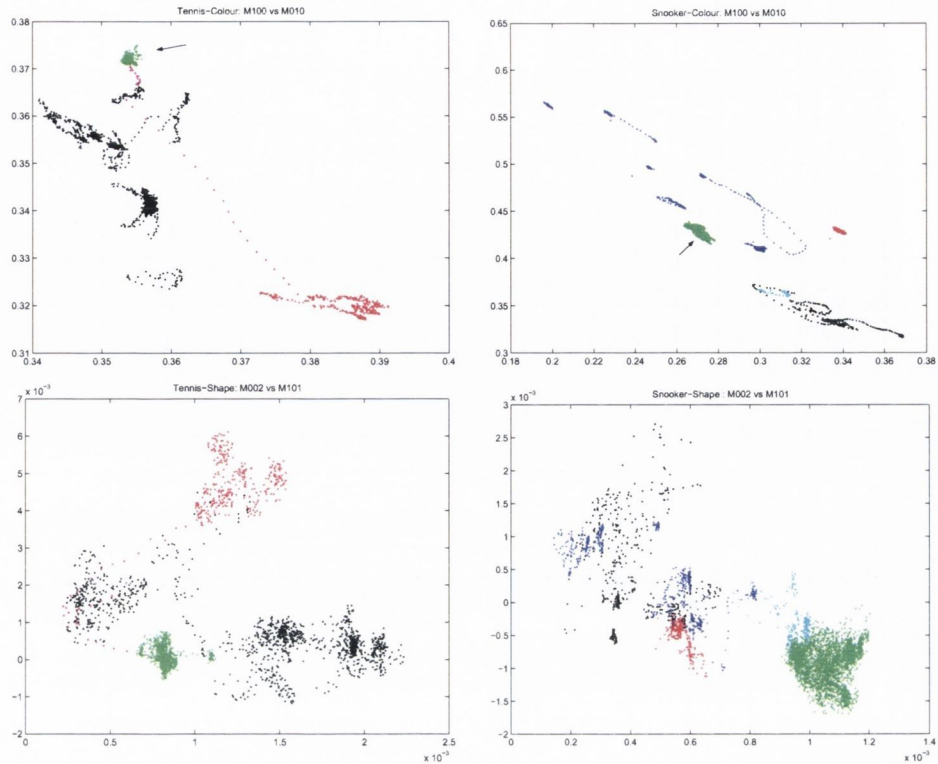


Figure 3.25: Plot of the statistical moment features: Top: Plot of chrominance information (r, g) for snooker (left) and tennis (right). The green clusters in each of the plots correspond to the global views in each of the sports; Bottom: Plots of the shape features M_{002} vs M_{101} for tennis (left) and snooker (right).

1. Shot cut detection
2. Global/Non-global view transition detection
3. Dissolve detection

Shot cut detection

Initial shot cuts are detected using the well known sum of absolute luminance histogram differences. Since histograms contain no information related to the spatial arrangement of pixels in the image, each frame is split into 5 segments. If the sum of luminance histogram differences for each local histogram exceeds a specific threshold, a shot cut is inferred. Figure 3.26 illustrates the arrangement of the local quadrants.

The sum of absolute luminance histogram differences between histograms $H_t(j)$ and

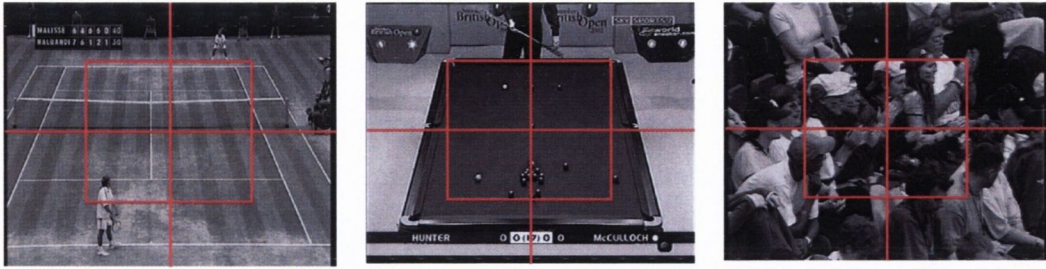


Figure 3.26: Quadrants for computing histograms over the image. The quadrants boundaries are shown in red.

$H_{t-1}(j)$ of m bins are computed for each quadrant, $\mathcal{Q} = (1 \dots 5)$.

$$d = \sum_{j=1}^m |H_t(j) - H_{t-1}(j)| \quad (3.26)$$

If d exceeds a threshold, $\tau_{cut}^{(\mathcal{Q})}$, for all quadrants, a shot cut is deemed to have occurred. An adaptive threshold based on the statistics of a window of 20 previous histogram difference values is used to set the thresholds for each quadrant. The mean, $\mu^{(\mathcal{Q})}$, and standard deviation $\sigma^{(\mathcal{Q})}$ are computed and the threshold is set as $\tau_{cut}^{(\mathcal{Q})} = \mu^{(\mathcal{Q})} + \beta\sigma^{(\mathcal{Q})}$. β is set to 5.

Global/Non-global view transition

Parsing sports footage according to the global or non-global view type is akin to detecting high-level shot cuts. This is because the geometry not only allows the shot cut to be identified but also the camera view and hence the importance of that shot for summary purposes. This immediately allows for exploitation of the context of these kinds of view type and could conceivably be a more powerful approach than the generic use of histogram based shot cut detection. For instance, in both tennis and snooker, shots of the crowd and of the players can be considered less important than shots containing game events which occur in the global view, so can be summarised simplistically, or discarded entirely. These high-level shot cuts can be inferred by searching the first frames of each of the detected shot cuts for the required geometry exhibited by a tennis court or a snooker table.

Dissolve detection

In order to detect dissolve transitions, a variation of the twin thresholding method outlined in Zhang et al [169] was implemented. It differs to [169] by dividing the image into the same quadrants used for the shot cut detection (figure 3.26). The method sets two thresholds based on the statistics of previous frames in the shot. The first threshold, $\tau_{cut}^{(\mathcal{Q})}$, is set to a higher

value which detects shot cuts, while the second threshold, $\tau_{dis}^{(Q)}$, of lower value initialises the dissolve detection.

If the lower bound is exceeded in more than three of the five segments, the difference between the low threshold and sums of absolute luminance histogram differences for each subsequent frame, are accumulated. If this cumulative sum is greater than the higher threshold and the current histogram difference is less than the low threshold in more than three of the five segments, a dissolve is inferred.

Figure 3.27 shows a plot of the histogram differences (blue) and cumulative histogram differences (red) for the middle quadrant in *Higgins*. The relevant dissolve frames from the footage are shown above the plot.

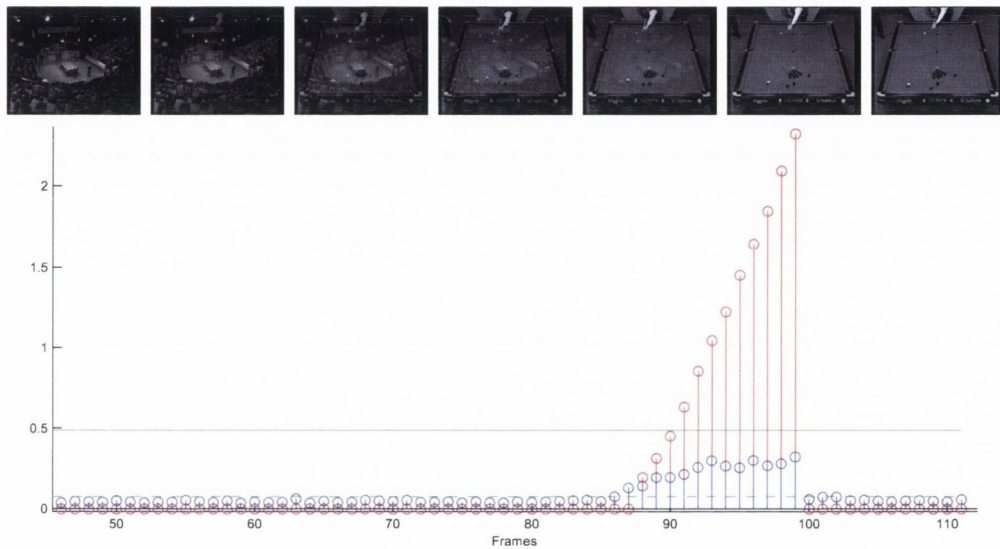


Figure 3.27: Dissolve detection. Histogram differences are shown in blue and the cumulative difference is shown in red. The dissolve is detected between frame 88 and 99. The low threshold is a dashed green line and the high threshold is solid green. The images shown are from frame 88 and every second one to frame 100.

3.5 Summary

This chapter introduced the first two steps in the proposed framework for sports video analysis. Feature extraction involved segmenting the playing surface and locating the delineating playing area. This was used to locate 'high-level shot cuts' in the footage. These types of shot cut detect the camera view as being global or non-global and hence understand the importance of that shot for summary purposes. The sequences of non-global views were tem-

porally segmented further into their constituent shots using conventional temporal boundary detection techniques.

A new feature which summarises the geometrical content of a scene without the need to calculate complex 3D geometries was presented. Additional colour and shape features which exploit the shape and colour content of each view were extracted from the footage which will be used to help classify the different view types.

4

Object Tracking¹

There has been a long history in the research of object tracking [17]. It has proved to be useful in surveillance applications (both in tracking of humans [90] and road traffic [83]), teleconferencing [154] and human-computer smart interaction [80]. Tracking can be difficult in the presence of clutter and generally relies on certain operator imposed constraints.

The ability to track objects in an image sequence is useful where the motion of an object, or several objects is important, and conveys useful information. This is particularly the case in sports where the motion of an object can embody the description of high-level events. In snooker, explicit tracking of the white ball from frame to frame can provide useful clues relating to the game semantics. For example, if the cue ball is struck, travels down the table, hits a coloured ball that is not potted and returns back to the baulk area, a conservative shot can be inferred. Similarly, if the white ball remains in the centre portion of the playing area, the player is deemed to be break building. Tracking a player around a court can allow certain types of plays to be recognised in tennis footage. A player moving from the base line to the net could mean that he is attempting a “serve-and-volley”

From the literature the tracking of objects can be divided into two classes:

1. Matching techniques: Matching techniques for tracking rely on segmenting the image into various components based on colour, motion and texture. The candidates are then matched to a specific template. Basic template matching techniques have been

¹Results from this chapter have been published as “Semantic event detection in sports through motion understanding” by N. Rea, R. Dahyot, and A. Kokaram in the *Proceedings of the 3rd International Conference on Image and Video Retrieval*.

used such as minimising the sum of absolute differences between the intensities of the candidates and target pixel areas [42]. A mean-shift matching method which deterministically searches for regions similar to a reference RGB histogram model has also been implemented allowing control over characters in first person perspective video games [21].

2. Probabilistic tracking: The general idea of probabilistic tracking (*e.g.* particle filters (PF), unscented Kalman filter (UKF), multiple hypothesis tracking (MHT)) is to evaluate several hypotheses and weight candidate models according to their similarity to a target model. The Kalman filter [156] is one such traditional probabilistic tracking method. It works by estimating a process state and updating the state with observations related to the state space. It is limited however, by its inability to handle non-linear state transitions and non-Gaussian process and observation noise.

Successful attempts of probabilistic object tracking in video have been implemented in a number of papers, using either edge/shape features [70], colour distributions [107,117] or a fusion of a number of features [118]. Particle filtering has proved to be a successful method of tracking objects in clutter [70,108] but can be used in most applications where the state of a system needs to be calculated as noisy observations become available .

In this chapter, probabilistic based approaches to tracking will be reviewed and the methods involved in particle filtering will be discussed. The implementation of a tracker based on the CONDENSATION algorithm will be presented and assessed using geometrical measures. For the applications considered in this thesis, improvements have been able to be made to the tracker. These include the use of:

- a) Likelihood ratios based on the colour distribution of the object to be tracked and that of the playing area.
- b) The use of Parzen windows for the estimation of the pdf of small objects.
- c) Using *a-priori* information from the geometry of the scene to scale the size of the target and candidate regions.

The results from the tracker will then be compared to those generated by a gradient based motion estimator.

4.1 Probabilistic tracking

The main objective in probabilistic tracking is the computation of a current hidden state, q_t , given current, X_t , and previous observations, X_{t-1} ². If the system is Markovian, the

²In the implementation of this object tracker, the states, q_t , are taken as the location of the ball at any time and the observations, X_t are the histograms of the samples in HSV (Hue, Saturation, Value) space.

evolution dynamics of the states can be defined using the function:

$$q_t = \mathcal{F}(q_{t-1}, \dots, q_{t-r}, a_t) \quad (4.1)$$

where $\mathcal{F}(\cdot)$ describes the state transition model, r is the Markovian order and a_t , the process noise. Noisy observations X_t arise from each state described by a function \mathcal{G} where b_t is the noise.

$$X_t \sim \mathcal{G}(q_t, b_t) \quad (4.2)$$

Under the Bayesian sequential framework, a posterior, $p(q_t|X_{1:t})$, can be approximated using a two step recursive process of prediction and updating. The goal then of tracking is to estimate a state, \hat{q}_t from the posterior which is sufficiently close to the true state q_t .

The two step Bayesian filtering approximation is listed below.

1. **Prediction:** In the first step, the prior for the next time step $t + 1$, $p(q_{t+1}|X_{1:t})$, is computed by propagating the posterior from the current time step t according to a transition density $p(q_{t+1}|q_t)$ (or the $\mathcal{F}(\cdot)$ function in equation 4.1).

$$p(q_{t+1}|X_{1:t}) = \int p(q_{t+1}|q_t)p(q_t|X_{1:t})dq_t \quad (4.3)$$

2. **Update:** Updating the posterior prediction is achieved by direct application of Bayes theorem, upon receiving a new observation X_{t+1} is given by the solution.

$$p(q_{t+1}|X_{1:t+1}) \propto p(X_{t+1}|q_{t+1})p(q_{t+1}|X_{1:t}) \quad (4.4)$$

The likelihood used to estimate the posterior is the function $\mathcal{G}(\cdot)$ given by equation 4.2.

Recursion of the two steps is however not generally possible. This is because, for a given state q_t , the observation likelihood model, $\mathcal{G}(\cdot)$, often produces observations which are non-linear and non-Gaussian as a result of non-Gaussian noise, a_t . Furthermore, non-linear/non-Gaussian state transitions often occur in practice affecting the transition function $\mathcal{F}(\cdot)$. The Bayesian solution to the posterior update also involves high dimensional integrals [43] whose solutions are generally analytically intractable.

This has motivated the foundation of approximations to the posterior, one of which is the particle filter (PF).

4.2 Particle Filtering

The main aim of particle filtering is to approximate a density using a discrete set of particles (or samples), $\{q_t^{(n)}\}_{n=1}^N$, randomly selected from state space. These particles have associated weights, $\{w_t^{(n)}\}_{n=1}^N$, based on a likelihood model. The approximated posterior can therefore be thought of as a randomly sampled weighted approximation of the true posterior, $p(q_t|X_{1:t})$.

The recursion equations from the previous section can be solved using Sequential Monte Carlo methods, a toolkit which is described at length in Doucet et al [44] and MacKay et al [96]. Under this framework, the previously intractable posterior, $p(q_t|X_{1..t})$, can be represented by a discrete set of N weighted samples. The discrete set of samples allows the integrals to be replaced by discrete summations. A derivation of the process is given in appendix C.

The sample set $\{q_t^{(n)}, w_t^{(n)}\}_{n=1}^N$ where w_t are the weights of each particle, are initially distributed according to a proposal function $u(q_t|X_{1..t})$. The proposal distribution is required as it is sometimes hard to sample from the true posterior. A mechanism to sequentially update the weights is given in equation 4.5 which is proved fully in Doucet et al [44].

$$w_{t+1} = w_t \frac{p(X_{t+1}|q_{t+1})p(q_{t+1}|q_t)}{u(q_{t+1}|q_{1..t}, X_{1..t+1})} \quad (4.5)$$

If the proposal distribution in the sequential update of the weights is chosen to equal the prior (the transition probability of going from a state at time t to that at time $t+1$), the new weight of each sample is directly related to the corresponding observation likelihood. This is also known as a bootstrap filter [43]. While not being the optimal proposal distribution, it is sufficient for low-dimensional spaces such as the colour likelihood model which is used here (see section 4.3 for specification of the likelihood function). The bootstrap approximation is therefore:

$$w_{t+1} = w_t p(X_{t+1}|q_{t+1}) \quad (4.6)$$

Although this update is easy to implement, it is possible that the majority of weights will eventually group around a local maximum. This is known as degeneracy. When this happens, it becomes difficult to approximate the posterior fully.

Sequential resampling of the weights is used to help avoid particle degeneracy (*i.e.* by retaining and multiplying samples of high likelihood and rejecting those with low likelihood), and is achieved by using what is commonly known as roulette wheel selection.

Roulette wheel selection entails mapping the approximation of the posterior $\{q_t^{(n)}, w_t^{(n)}\}$ into an equally distributed measure $\{q_t^{(n)}, 1/N\}$ by generating a random number $r \in [0, 1]$ and selecting the smallest sample n such that cumulative sum of samples up to n is less than r .

The entire process can be described by the following steps and in figure 4.1.

1. **Prediction:** Perturb the particles according to a deterministic drift and an individual zero mean stochastic component. In the case of snooker ball tracking the drift is a second order AR motion model with a stochastic component $\epsilon \sim \mathcal{N}(0, \sigma)$. Tennis uses the stochastic component alone. This will be discussed in section 4.3.
2. **Measurement/Update:** Calculate the likelihoods of the samples and calculate the new weights according to equation 4.5.

3. **Sample/Resample:** Select N samples based on their weights according to $r \in [0, 1]$. This will select multiple samples with high probabilities. The samples are initialised with equal weights.

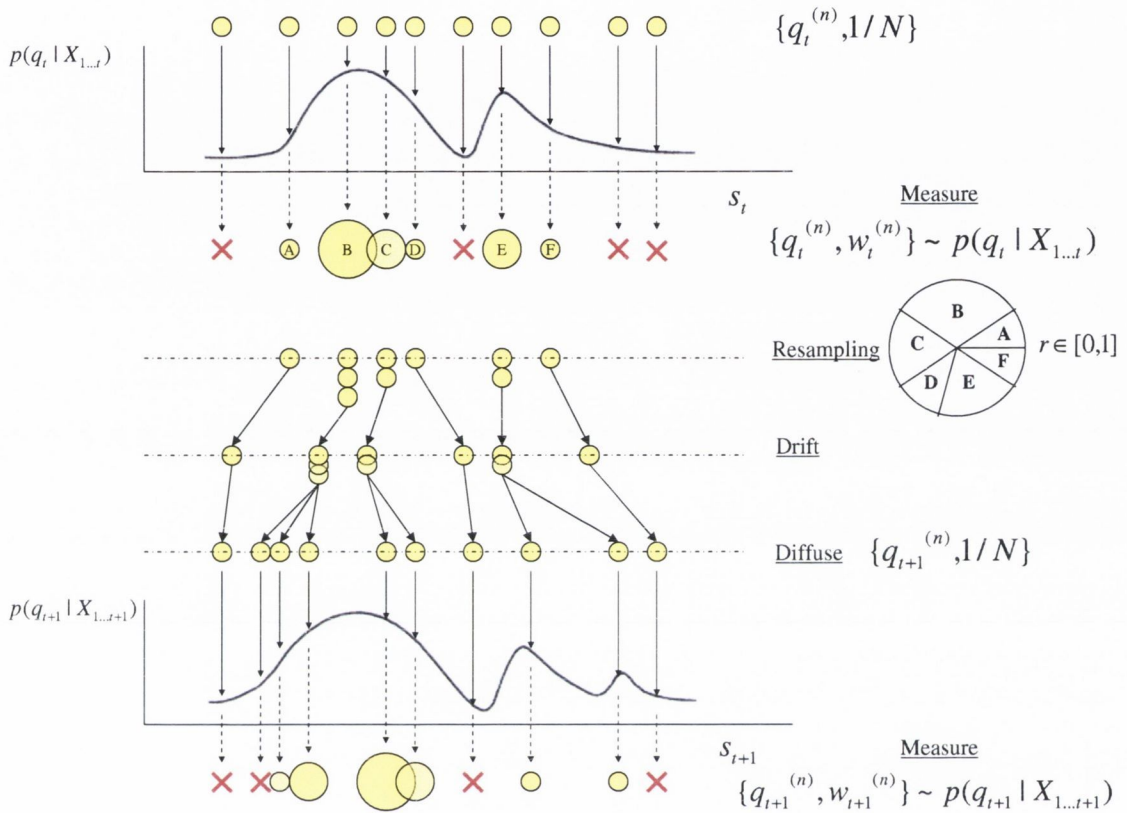


Figure 4.1: Particle filtering for one iteration from time t to $t + 1$.

4.3 Generic implementation of the tracker

Edge based image features have been traditionally used for contour tracking under a particle filter framework [70, 90]. In snooker however, the edges of the balls are not always clearly defined. For example, if two balls of similar colour are beside each other it may be difficult to distinguish the two individual balls by shape alone. Furthermore, motion blur causes the perceived ball shape to become elongated. This motion blur occurs instantaneously when a ball is hit from its initial resting state.

With regard to the tennis footage, creation of a tennis player edge model could also prove to be extremely difficult. Thus far, tracking of edges has been limited to models of head and shoulders of humans and objects of relatively simple geometry such as leaves and cars [70]. Players on the tennis court move about vigorously as they attempt to hit the ball while also

deforming due to motion blur. In the case of both tennis and snooker, the quality of the captured footage also contributes to the difficulty in using edges to track the objects. A colour based approach is therefore adopted for object tracking in both snooker and tennis footage.

HSV (Hue, Saturation, Value) space colour histograms are used to approximate the colour distribution of the objects and create a target and candidate models for computing the particle likelihoods. *HSV* space is used because it allows separate histogram comparisons by decorrelating the brightness and chrominance components.

Histograms offer the properties of being scale and rotationally invariant and robust to partial occlusion. While rotation invariance is not an issue for tracking in this application, the first and last properties of colour histograms are particularly useful for tracking snooker balls and tennis players. Furthermore, it is trivial to impose a weighting function thereby giving more importance to pixels in certain locations. The weighting function, z , used for snooker and tennis footage is given in sections 4.4.2 and 4.5.2. The target model is generated from an automatically selected object region from the first frame of the sequence in which the object is to be tracked, and is retained throughout the shot. It has been shown that this approach can achieve robust tracking even if there is deformation of the shape of the object being tracked [28, 117].

As the objects in snooker and tennis move in the vertical plane of a camera view which enforces true perspective, they are subject to changes in scale due to perspective distortion. Candidate regions must therefore be scaled appropriately to ensure that the presence of background pixels in the colour distribution is minimised. Knowing the physical dimensions of the playing surfaces (the dimensions of the snooker table along with a schematic of a tennis court in appendix B), an approximation of the physical size of the objects and their current co-ordinates along with the perceived length of the lines in the image (obtained from sections 3.3.2 and 3.3.1 for tennis and snooker respectively), the size of the object in pixels can be approximated by analysing the perspective distortion of the playing surface. The object sizes are scaled according to the proportional reduction in playing area width at the location of the object relative to the reduction in length of the top delineating line with respect to the bottom line. Other tracking applications do not have this knowledge to hand and instead rely on IIR filtering of the similarity between candidates and target models for a preset increase or decrease in region size [28].

4.3.1 Establishing the likelihood model

A HSV colour histogram of a circular region is used to construct the likelihood of the particle regions in snooker. A rectangular region is used for the tennis footage. The idea is that the likelihood should encourage the matching of regions with similar colour distributions. A Bhattacharyya distance measure [55] is used to calculate the similarity between candidate

histograms, ρ , and the target, ξ , and is in turn used to weight the sample set, $\{(q_t^{(n)}, w_t^{(n)}) | n = 1 \dots N\}$. The sample likelihoods are computed as

$$p(\rho_t^{(n)} | q_t^{(n)}, \xi) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(1-h[\rho(q_t^{(n)}), \xi])}{2\sigma^2}} \quad (4.7)$$

Where,

$$h[\rho, \xi] = \sum_{j=1}^m \sqrt{\rho^j \xi^j} \quad (4.8)$$

where ρ is the candidate histogram, j is the j^{th} of m histogram bins and ξ is the target or prototype model.

To further improve the tracking, prior knowledge of the playing surface can be incorporated into the weighting of the candidate regions. This has been used to good effect in face and object detection applications [12,78]. In these applications it was shown that a likelihood ratio between face models and non-face models can help reduce the number of false alarms. This principle is applied to particle filtering where a ratio of likelihoods of each candidate given the object model, and the likelihood of the candidate given a playing surface model, is calculated.

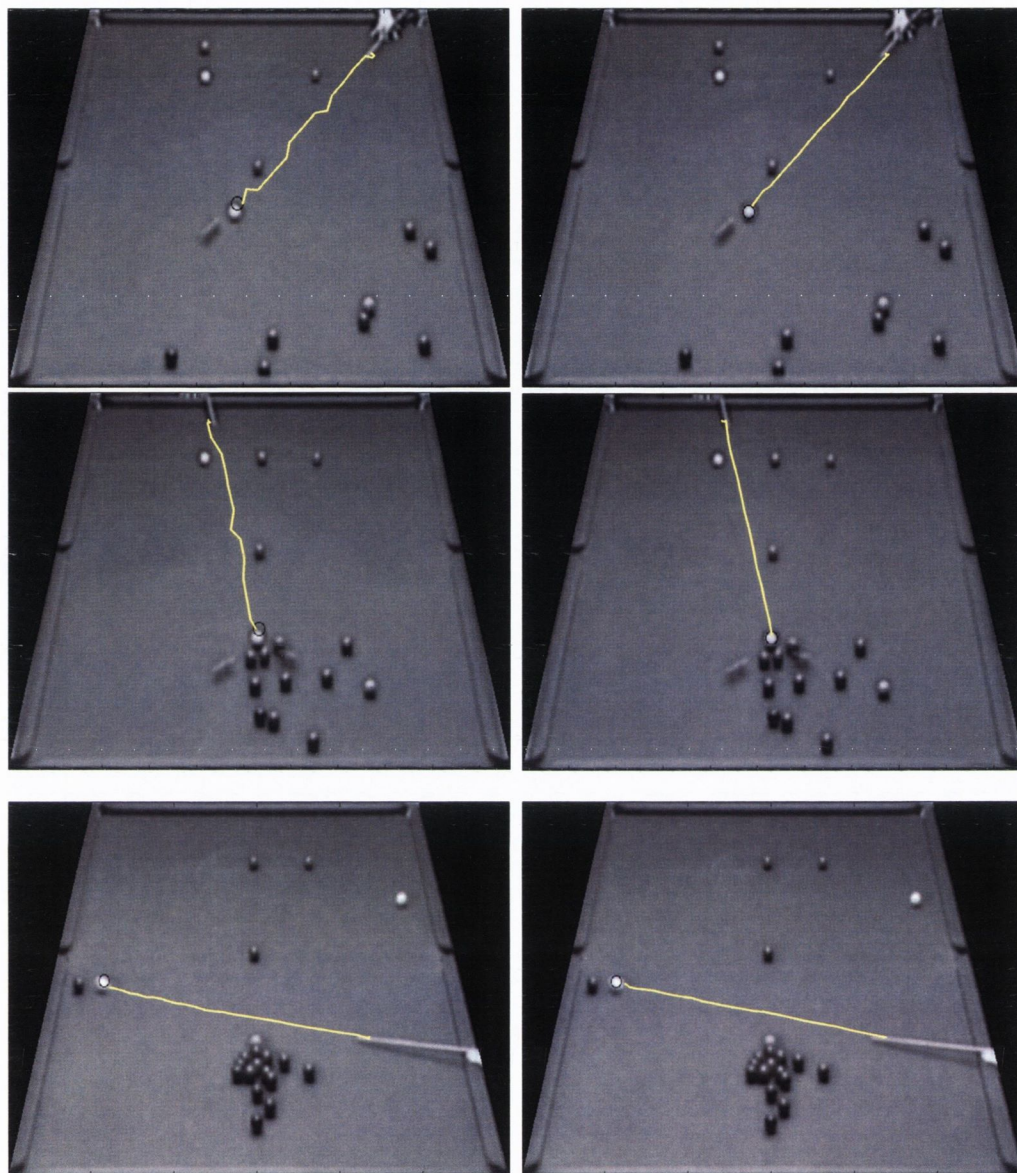
A colour model for the playing surface, κ , (*i.e.* the snooker table and tennis court) can be trained by manually selecting regions of the surface from each footage source. This is done in a similar fashion to that used for estimating the colour pdf of the playing surface for segmentation using the Gaussian mixture model (GMM) in section 3.2.3. The likelihood of each candidate region having been generated by the playing surface model, κ , can be computed using the same likelihood as equation 4.7 with ξ being replaced by κ .

A likelihood ratio, $\chi_t^{(n)}$, of object to non-object regions can be calculated using equation 4.9.

$$\chi_t^{(n)} = \frac{p(\rho_t^{(n)} | q_t^{(n)}, \xi)}{p(\rho_t^{(n)} | q_t^{(n)}, \kappa)} \quad (4.9)$$

Tracking using likelihood ratios gives better tracking fidelity as it encourages tracking of the selected object and not regions with a large number of playing area pixels, forcing the particle to be more centred on the object to be tracked. This is particularly useful in the snooker footage. When the ball is hit with a great deal of force, it is perceived to have become elongate due to the slow frame rate of the camera. The colour distribution of the entire object changes with some regions appearing to be an amalgamation of some table pixels and some ball.

An illustration of this is shown in figure 4.2 where the white ball has been tracked over 15 (top), 10 (middle) and 20 (bottom) consecutive frames respectively. As can be seen from the image on the top right, the track produced using the likelihood ratio is smoother, and a collision is detected. The middle row shows the white ball being hit, again with great force. The track produced using likelihood ratios is, once more, superior to that produced by the



(a) Object likelihood

(b) Object and background likelihood ratio

Figure 4.2: Comparison of tracking: Tracking the white ball using the ball colour likelihood and the table and ball colour likelihood ratio. In cases where the ball is hit with a great deal of force by the player (top and middle rows) the tracking produced using likelihood ratios is superior to that of the likelihood based on ball colour alone.

object model alone. The tracks produced by both methods in the figures on the bottom row illustrate the comparison of trackers for a white ball that is moving relatively slowly. In this case, both methods produce equally good tracks as the ball colour is not significantly distorted as a result of motion blur.

4.3.2 Establishing the proposal distribution

It has been established that the optimal proposal distribution is the posterior [43], but since the posterior is not generally available, a new proposal must be created from auxiliary features. The objective of this distribution is to relocate particles to areas of high posterior probability by taking into account the previous state and current observations from an auxiliary source. In Perez et al [116], audio visual features are used to estimate prospective regions of high posterior probability. Motion and audio are used under separate conditions to simulate a proposal distribution and state space particles are generated from these measurements. These particles are passed to a colour based particle filter which refines the search. For example, regions from the previous state which exhibit high motion activity are given a large weight, so particles which are resampled for colour filtering should already be in good locations in state space to provide a reasonable approximation to the posterior distribution. Motion is a particularly helpful proposal when occlusion is present in the sequence. Using the motion as a proposal, the tracker can be reinitialised if the target is lost at any stage. While partial occlusion can be a problem in snooker, this can be dealt with using the technique outlined in section 4.4. Full occlusion is not generally problematic in snooker or tennis as the objects are almost always in the frame³. If tracking is lost the object should reappear in, or close to, the position from which it left.

Audio can also be used as proposal. A stereo microphone array can be employed to estimate the horizontal regions in which a speaker might be located. The particles generated from this proposal are then passed to the colour filter which then refines the search [118]. Audio is only useful when the layout of a scene is known *a-priori* (such as for the application of tracking talking heads in video conferencing where audio features parameter can be configured).

For the tracking in this thesis, the bootstrap implementation of the particle filter is used. It specifies the proposal distribution as the prior (or the transition distribution $p(q_{t+1}|q_t)$). So the posterior takes the form:

$$p(q_{t+1}|X_{t+1}) \propto p(X_{t+1}|q_{t+1})p(q_{t+1}|q_t) \quad (4.10)$$

As discussed, this prior is generally a weak motion model with a stochastic component which, by its nature, only makes use of previous state information. The implementation of this particle filter employs a two step iterative prior which updates the prior by recycling

³This may be useful in doubles tennis but this kind of footage is not considered in this thesis.

particles with high weight. If the cumulative likelihood of all the particles exceeds a threshold the object is deemed to have been found and the particles are perturbed according to a motion model. The process is outlined below

- **Iteration Prior**

The prior used during the iterative process recycles relevant particles based on their individual weights. If the cumulative likelihood of all particles is not sufficient to assume a correct lock, particles with high likelihoods are kept and the remaining particles are redistributed according to a contracting-expanding algorithm. This process is iterated on each frame until a good estimate to the posterior is achieved.

As is typical for object tracking, it is necessary to distinguish between correct tracking in the next frame and loss of 'lock'. This can be detected by using a threshold on the sum of the particle likelihoods, L_τ . If the condition in equation 4.11 is fulfilled, a correct lock is assumed and the ball is deemed to have been found in frame t where L_t is the cumulative sum of the particle weights.

$$L_t = \sum_{n=1}^N \chi_t^{(n)} \geq L_\tau \quad (4.11)$$

The minimum mean square error (MMSE) estimate is taken as the current position of the object and is calculated using equation 4.12.

$$\mathcal{E}[q_t | X_{1..t}] = \sum_{n=1}^N q_t^{(n)} \chi_t^{(n)} \quad (4.12)$$

Where q_t are the positions of the particles and the posterior is the weight on each one (*i.e.* $\chi_t^{(n)}$).

If the cumulative likelihood of the samples is less than the threshold ($L_t < L_\tau$), the ball is deemed not to have been found and a new prior is used for the next iteration. Particles with high likelihoods are kept, and those with low likelihoods (< 0.01) are perturbed using an iterative expanding-contracting particle distribution method. The purpose of this is to increase the range of the particle filter if the original prior does not give a good lock (*i.e.* if the object has moved in a way such that it is out of scope of the tracker). This does not impinge on the validity of the PF process since it is only a superficial method of improving a lock using low-likelihood particles.

In each step, if the likelihood produced by a single particle is greater than that of the most likely individual particle in the retained set, the new particle (that of greatest likelihood) is used as a seed for the next search (assuming $L_t < L_\tau$) and the expanding-contracting process is reset. The contracting expanding method is outlined as follows

1. **Contract:** If $L_t < L_\tau$ from step 2, the relevant particles at iteration i , $q_{t_i}^{(l)}$, are propagated according to the prior. The zero mean Gaussian with variance of 2 is chosen for a tight spread of particles.

$$p(q_{t_i}^{(l)} | q_{t_{i-1}}) = \mathcal{N}(0, 2), \quad i > 0 \quad (4.13)$$

2. **Expand:** If $L_t < L_\tau$ from step 1, distribute the relevant particles according to the motion model (equation 4.16) with the stochastic component:

$$p(q_{t_i}^{(l)} | q_{t_{i-1}}) = \mathcal{N}(0, \sigma^2) \quad (4.14)$$

where $\sigma^2 = 4$ but is incremented by one on each successive iteration.

The Maximum a Posteriori (MAP) estimate 4.15 is used as an approximation for the position of the object if the particle filter does not converge after 7 iterations.

$$p(q_t | X_{1..t}) = \arg \max_{1 \leq n \leq N} [\chi_t^{(n)}] \quad (4.15)$$

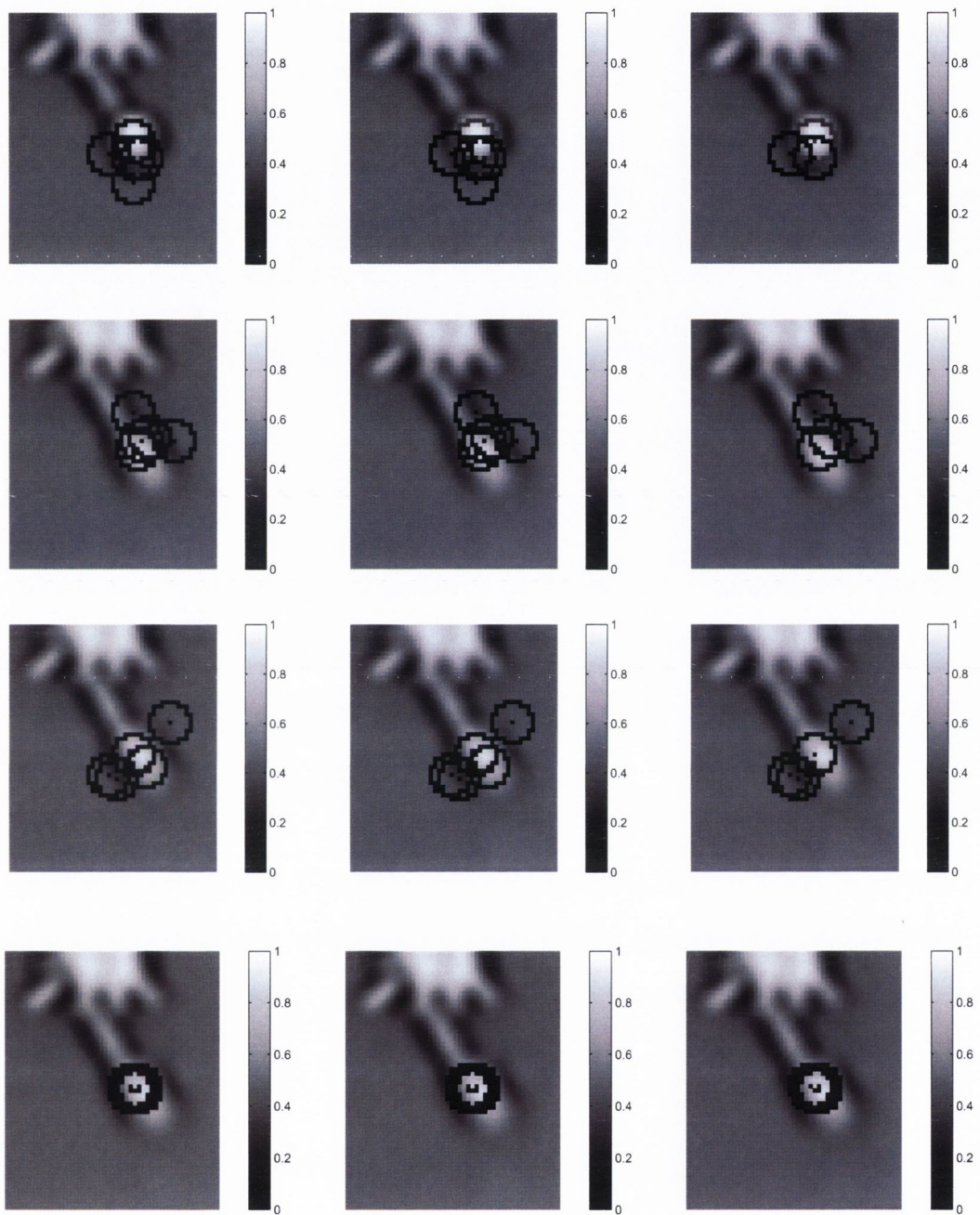
An example of the particle distribution is shown in figure 4.3 where the process of tracking the white ball from snooker footage from its initial stationary position to the first frame of its motion is illustrated. The colour bar shows the likelihood of each particle. The tracking is rather coarse as only five particles are used. This is simply for illustrative purposes⁴. The process is initialised by giving each particle equal weight. The weight on each particle is then measured according to the likelihood given above (equation 4.9). Appropriate particles are then chosen by roulette wheel selection. These particles are then perturbed by the motion model. Three stages of initial weighting (column 4.3(a)), likelihood calculation (column 4.3(b)) and roulette wheel selection (column 4.3(c)) are shown in figure 4.3.

In figure 4.3, the top row shows the first frame of the sequence (full resolution frame shown in figure 4.4 (left)). The second to fourth rows show the first to third iteration of finding the white ball in the second frame (full resolution frame shown in figure 4.4 (middle)). A subsequent full resolution frame is shown in figure 4.4 (right). Iteration two of frame two shows the expanding of the search region and iteration three is the contraction, finding the ball.

• Transition Prior

On each successful iteration of the tracker, the state of particle is recorded and the particle set is dispersed. For snooker, assuming linear motion, this is taken to be a second order auto-regressive motion model which places more emphasis on motion from the previous frame than the frame before that (*i.e.* $p(q_{t+1} | q_t \dots q_1) = p(q_{t+1} | q_t, q_{t-1})$).

⁴For experiments conducted in later sections, 100 particles are used.



(a) Initial weighting

(b) Likelihood calculation

(c) Roulette wheel selection

Figure 4.3: *Tracking of the white ball with 5 particles.*

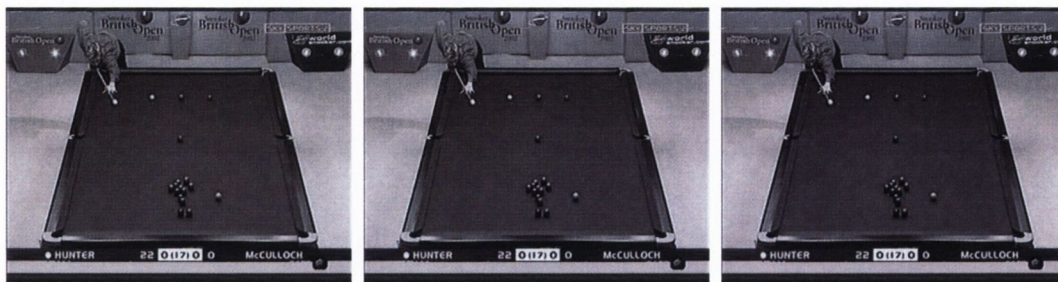


Figure 4.4: Three full resolution frames from the video sequence used in figure 4.3.

Each sample is then perturbed according to the stochastic component of the motion model as shown in equation 4.16.

$$q_{t+1} = q_t + [\alpha (q_t - q_{t-1})] + [(1 - \alpha) (q_{t-1} - q_{t-2})] + \epsilon \quad (4.16)$$

where $\alpha = 0.7$, and $\epsilon \sim \mathcal{N}(0, r)$. The same process is used for both horizontal and vertical directions.

Tracking the tennis player is a more difficult problem than that of tracking the snooker ball. Due to the presence of global motion and the pathological motion undergone by the player, such motion models used for the snooker ball tracker do not hold. This is the case for tennis footage where there is generally a large amount of horizontal translational global motion. Assuming that the player always tries to be near the ball, a prior with just a stochastic component is sufficient to locate the player. In any case, the iterative prior helps refine the search for the object to be tracked.

4.4 Tracking snooker balls

As was discussed in section 4.3, a target model of the colour distribution of a snooker ball is created in the first frame of the clip from an initial location. A snooker clip is defined as the instance at which the cue ball is first set in motion until the time at which it, and all other balls being tracked come to rest. Initialisation of the colour model can be applied manually or by means of the method described in section 4.4.1 for tracking the white ball. Coloured balls which have been in collision with the white ball can also be tracked by analysing the velocity of the cue ball (section 5.2) and instantiating a separate track for these balls.

4.4.1 Localisation of the white ball

It is important to accurately model the colour distribution of the white ball for correct tracking because the evolution of the location of the white ball from frame to frame provides



Figure 4.5: Binary maps. Left-to-right: $(V(i, j) - R(i, j)) \leq 0$; $(V(i, j) - S(i, j)) \geq 0$; $((V(i, j) - R(i, j)) \leq 0 \vee (V(i, j) - S(i, j)) \geq 0)$.

high level information about the type of shot being played. A target colour distribution is established by locating the white ball at the start of each clip of the full table. Localisation of the white ball is accomplished using a combination of segmentation based on thresholding and detection of a bright moving region on the table by frame differencing.

The player must first be removed from view because moving white components attributed to his attire might interfere with correct detection of the white ball. A binary map of the player (figure 4.5), $player(i, j)$, is created by thresholding the colour plane differences below (equation 4.17), where V is the brightness from HSV space, R is the red and S is the saturation components.

$$player(i, j) = ((V(i, j) - R(i, j)) \leq 0 \vee (V(i, j) - S(i, j)) \geq 0); \quad (4.17)$$

By applying this segmentation without consideration of the player's location, a number of balls will also be inadvertently masked. The player is distinguished from ball objects on the table by finding the largest region that is connected to the edge of the table (recalling that the table edge has already been located using the methods outlined in chapter 3).

In order to remove the player from the view, the detected player region must be filled with some suitable table information. From the centre of the table, the gradient of the intensity of a region of size 30×30 pixels is calculated $[I_x, I_y]$ (defined in section 3.3.4). If the sum of the gradient magnitude over the region is less than a specific threshold, $\tau = 30$, a flat area of the table is deemed to have been found (equation 4.18). Otherwise the region is shifted toward the top of the table, which is generally less densely populated by balls, until a flat region is found.

$$\sum_i \sum_j \sqrt{I_x^2(i, j) + I_y^2(i, j)} < \tau \quad (4.18)$$

False masking of coloured balls, either as a result of being too close to the tables edge or to the player, is not considered to be too costly as correct detection of the white ball is all that is required. Results of the player segmentation and masking are shown in figure 4.6.

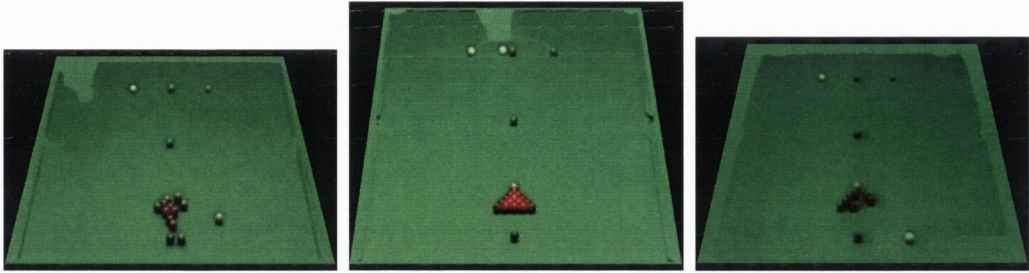


Figure 4.6: *Player masking. Left to right: Hunter, Higgins, Hendry footage. In the Hunter footage, the player's cue is very close to the white ball. The algorithm considers it as being part of the player region. The black region surrounding the table in each image was computed using the table finding technique outlined in section 3.3.1. This allows detection of suitable regions that are connected to the table.*

Frame differencing is used to uncover any motion that may have occurred between frames, indicating that the cue ball was struck by the player. A problem arises using this form of differencing. As the player walks around the table he may occlude balls as he passes. These balls will be masked in subsequent frames (depending on the speed of the player), as the player masking algorithm considers them as being a part of the player. The balls will then suddenly reappear as he continues around the table. This will manifest itself as impulsive motion in the frame differencing binary map. A further thresholding on the colour planes is therefore required to determine if the moving region is white.

For all 'moving' object detected by frame differencing, windows of $6r \times 6r$ pixels (where r is the radius of the ball) centred around the mean location of objects are selected. By applying a threshold to the non-masked frames (equation 4.19) on the intensity and saturation of these windows, a binary map corresponding to white objects can be found. If an object of size less than $\frac{3}{2}\pi r^2$ and greater than $\frac{1}{2}\pi r^2$ is found, it is deemed to be the window containing the white ball, $whiteball(i, j)$.

$$whiteball(i, j) = (V(i, j) > 160) \wedge (S(i, j) < 140) \quad (4.19)$$

The white ball localisation algorithm iteratively back-tracks to find a frame where the magnitude of frame differences is lowest. This will be the frame in which the white ball is stationary. If the distance between the centre of gravity of the segmented white ball objects is less than 5 pixels, the stationary white ball is deemed to have been found. The target colour model must be computed while the white ball is stationary, as motion artefacts will corrupt the model histogram.

Experiments conducted on 30 shots from the footage sources where the white is hit by the player results in 100% correct localisation of the white ball.

4.4.2 Tracking the balls

A *HSV* space colour histogram, of a circular region specified by the relative size of the ball in relation to the table, is calculated. As a snooker table can be affected by luminance gradients due to non-uniform lighting conditions, the brightness component of the colour space was quantized to 16 bins. The colour histogram was represented using a concatenation of 3 separate 1-D histograms of (256+256+16) bins. Figure 4.7 shows the individual H, S and V histograms of the selected white ball region shown in black. The goal of the particle filter is to try and match the hypothesised model with the target model.

A collision between balls or a collision between a ball and the bottom cushion of the table may temporarily block the ball being tracked from view. Therefore, partial occlusion of the ball must be addressed. These peripheral pixels are unreliable when attempting to calculate the colour distribution of the object. Hence, a kernel with a monotonically decreasing profile from the centre of the object to its extrema assigns a lesser weight to those pixels. This is done in both the calculation of the colour distribution of the target and candidate models. It also proves useful for avoiding the incorporation of the colour properties of the table into the ball model. The weighting function is given in equation 4.20. For a ball with a radius of 7 pixels, the corresponding pixel weighting is shown in figure 4.8 where $r = \|\mathbf{x}_i - \mathbf{x}_c\|$.

$$z(\mathbf{x}_i, \mathbf{x}_c) = 1 - \left(\frac{\|\mathbf{x}_i - \mathbf{x}_c\|}{\max_i(\|\mathbf{x}_i - \mathbf{x}_c\|)} \right)^2 \quad (4.20)$$

Depending on the angle of orientation of the camera in the global view and the amount of space taken up by the table, the ball object normally varies in size from 5 pixels to 7 pixels in radius. This is typically not enough data to empirically yield a useful histogram. The use of Parzen windows resolves the problem of sparse data by spreading the distribution. The noise σ , is computed over a region of 30×30 pixels within the bounds of the table, exhibiting sufficiently low gradient, in the same way as computing the texture for player masking (section 4.4.1). The Parzen window is not applied to the brightness component as it has been quantised to 16 bins. As a result of using such a coarse bin quantisation there should be sufficient data to yield a good representation of the colour distribution.

The colour space is represented by $\Psi = \{H, S\}$. The effect of the Parzen window on the hue and saturation components of the white ball are shown in figure 4.9. The colour distribution, $\rho = \{\rho^u\}_{u=0\dots m-1}$ of the object region, R , is given as:

$$\rho^u = c \sum_{\mathbf{x}_i \in R} z(\|\mathbf{x}_c - \mathbf{x}_i\|) \phi(j - \Psi(\mathbf{x}_i)) \quad (4.21)$$

Where c is a normalising factor, \mathbf{x}_c is the location of the centre of the ball, $j = [0\dots m - 1]$ and where z is given in equation 4.20 and where $\phi(x)$ is the Gaussian kernel given by equation 4.22.

$$\phi(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \quad (4.22)$$

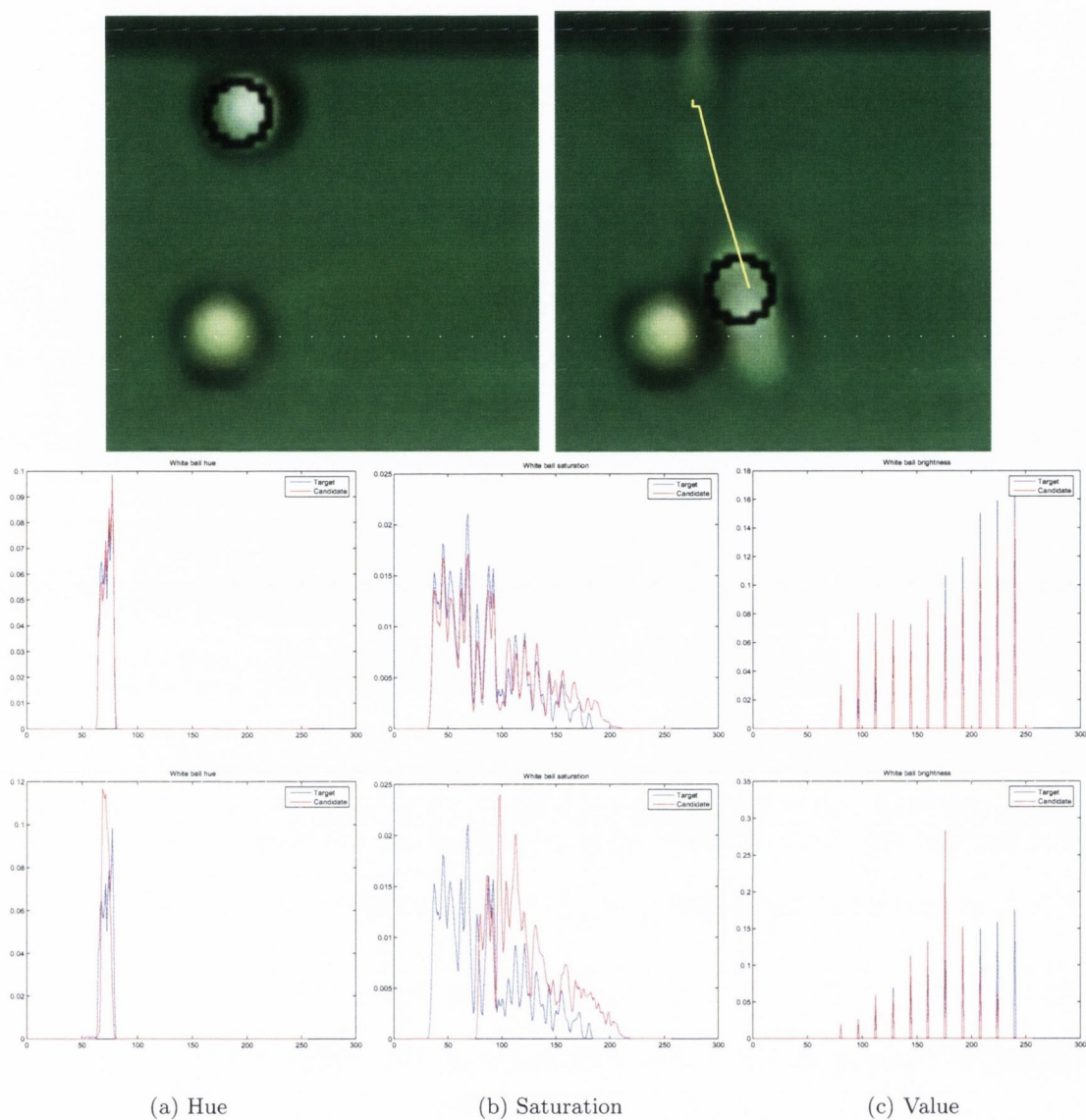


Figure 4.7: Target histograms and candidate histograms of the cue ball at frame k and frame $k + 5$. Top row: White ball at frame k (left) and $k + 5$ (right); Middle row: Target (blue) and the weighted candidate (red) histograms (H , S , V) of the white ball in frame k ; Bottom row: Target (blue) and the weighted candidate (red) histograms (H , S , V) of the white ball in frame $k + 5$.

4.5 Implementation of tennis player tracker

A slightly modified version of the tracker outlined in section 4.3 is used to track the player in the tennis footage. As previously discussed, the global view is considered to be the camera

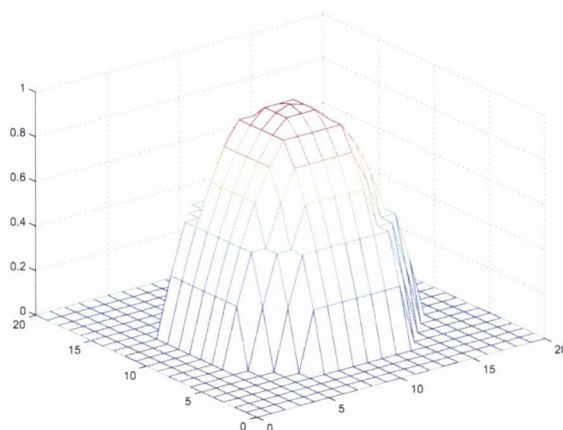


Figure 4.8: Ball pixel weighting $z(r)$

view which conveys the most information to the viewer. In this view the position of the players can be tracked from frame to frame.

4.5.1 Localisation of the player

The tracker is initialised by finding the player regions in the bottom and top halves of the court. This is achieved using a greedy algorithm on the brightness and saturation histograms for the player in the top half and the histogram of hue and brightness colour spaces for the player in the bottom half. The greedy algorithm has been discussed in section 3.2.1.

The clothing that the players wear tend to exhibit quite contrasting colours to those of the court surface. The white colours worn by the players are generally of high brightness, low saturation and high hue. The hoardings behind the player in the top half also show evidence of high hue, so this cannot be used to extract the player on that side of the court. For the player in the top half of the court, values greater than the range of brightness returned by the greedy algorithm and values less the 'greedy range' from the saturation component are considered not to be attributed to the playing field. These are labelled as player regions. Relevant values greater than the maximum and less than the minimum values of 90% range of the greedy histogram are sought. This is because, as tournaments progress, particularly on grass surfaces, regions tend to wear down and become notably brighter. The clothing worn by the player is both brighter and less saturated than these regions.

The player on the bottom half of the court is segmented using the brightness and hue component. Values greater than the maximum value returned by 90% of the range of the greedy histogram for both colour spaces are used to detect this player.

Any court lines that have been detected using the segmentation can be suppressed using the court finding technique outlined in chapter 3.3.2. Furthermore, the net area is masked by

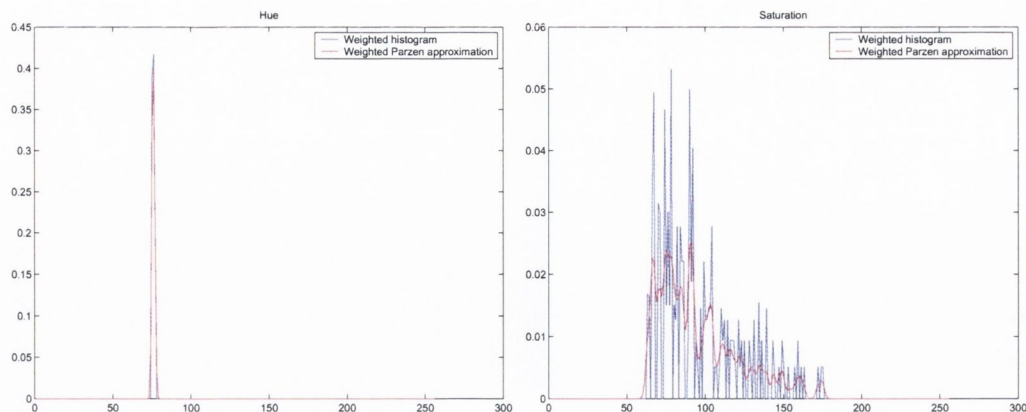


Figure 4.9: *Weighted histograms and the corresponding Parzen approximation of the hue and saturation components of the white ball. The red plot is the Parzen approximation and the blue is the weighted histogram of the ball using equation 4.20.*

approximating the height of the net at the centre point using the same technique as for the player scaling. The entire region across the net is masked to eliminate any ball boys that are close to the net. Areas below the bottom of the net region and above the top of the net are dilated because the lines that have been masked may also have masked some of the player.

The centre of gravity of the ‘blobs’ (which correspond to the torso of the players) of greatest size, in the bottom and top halves of the court are computed. It was found heuristically that the centre of gravity of a player can be estimated by shifting the centre of gravity by $l/4$ beneath the original value, where l is the maximum height of the blob. Using this value as the centre of the region to be selected and assuming that the average height of a player is 1.80 meters and has a width of 0.5 meters, the region can be scaled relative to the perspective distortion of the tennis court (using the same method as that used for scaling the snooker balls in section 4.4) such that it sufficiently frames the tennis player. The dimensions of a championship size tennis court are shown in figure B.2 in appendix B. Localisation of the player in this fashion is shown for all grass court footage in figure 4.5.1.

Initial player detection experiments conducted on 30 shots of the global view from the four sources of tennis footage results in 100% correct localisation for the top and bottom players.

4.5.2 Tracking the player

The HSV colour space is once again used and a colour histogram of the player region is calculated. Full 8 bit colour histograms (256+256+256 bins for H, S, V) were used to approximate the distributions in this case. A Gaussian, $z \sim \mathcal{N}(0, \mathbf{R})$, is used to weight the pixels within

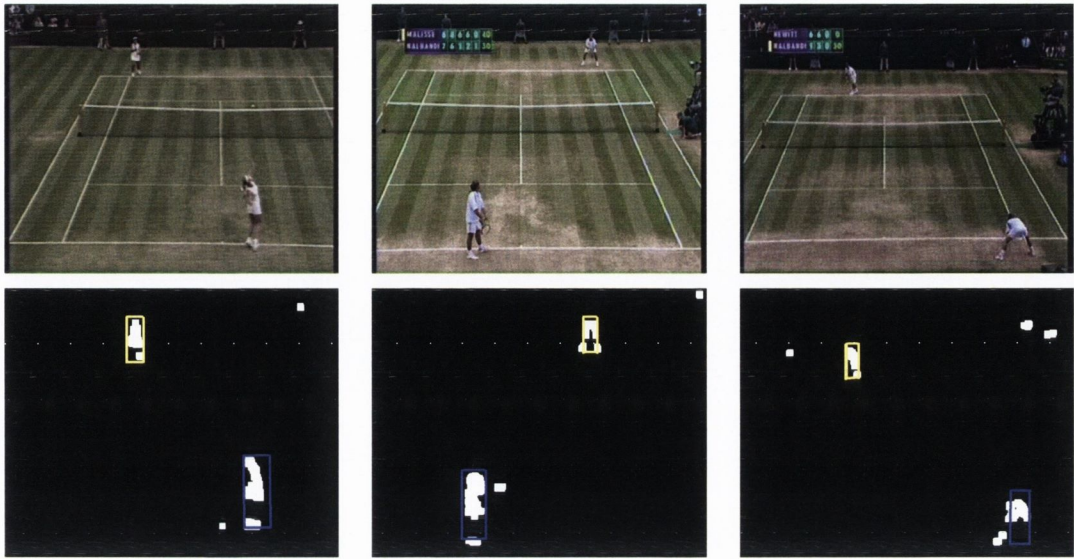


Figure 4.10: *Player localisation. Top: Original images; Bottom: Segmented player using greedy histogram segmentation and court segmentation. The binary map of the bottom and top players have been concatenated to form this image.*

the candidate and target regions. This weighting kernel gives a higher importance to pixels in the centre of the window and less to those pixels at the edge. The variances are related to the size of the window given by:

$$\mathbf{R} = \begin{pmatrix} \frac{h}{6} & 0 \\ 0 & \frac{w}{3} \end{pmatrix} \quad (4.23)$$

where h is the height of the player in pixels and w is the width of the player. A 3-D plot of the kernel is illustrated in figure 4.11 where the height and the width of the player region are taken as $h = 150$ and $w = 75$ respectively.

A Parzen approximation was not needed for tracking the tennis player as the size of the region is sufficient to yield a useful histogram. A playing area model is constructed by preselecting several regions of the court off-line. A likelihood ratio can then be formulated as outlined in section 4.3, and the players can be tracked around the court.

Figure 4.12 shows the bottom player located at two frames in the footage, the second of which is 50 frames after the first. The target and candidate histograms of the player region are shown for each frame, where the blue in each plot is the target and the red is the weighted candidate. Six stills from a clip of 500 frames of the sequence *Hewitt* are shown in figure 4.13 where both the top and bottom players are tracked.

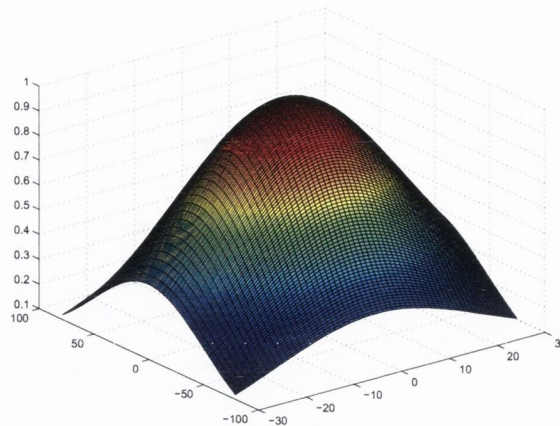


Figure 4.11: Tennis player weighting, z .

4.6 Assessment of the performance of the particle filter

In order to evaluate the performance of the tracker, the tracks obtained from the snooker footage were assessed using two geometrical measures. As the snooker balls generally travel in a straight line (assuming they are hit without any side spin) until a cushion or other ball is hit, it is possible to measure the deviation of each location estimated by the particle filter from the true trajectory of the ball.

In total, 11 shots made by the player (5 red, 2 black, 2 blue, 1 brown, 1 pink and 1 green balls) were assessed. These occurred at different stages in the game and in various locations on the table from the different sources of snooker footage. Two of these tests were conducted on balls which were potted. The velocity of the balls also varied. In total, the tests represented analysis on approximately 300 frames. Two performance measures were used to assess the performance of the tracker. In both cases the true trajectory is taken as a straight line between the starting and ending motion positions of the ball. The true trajectory of four shots is illustrated in figure 4.14. The tracks achieved using the particle filter are also shown.

4.6.1 Perpendicular distance from points

The first measure is the length of the normal, d_i , from the true trajectory, $y = mx + c$, of the ball to the projected position of the ball (x_i, y_i) . It is assumed that the true trajectory is the line connecting the initial position of the object to its final position. The perpendicular distance from a point to the true trajectory is described in [98] as:

$$d_i = \frac{|y_i - (c + mx_i)|}{\sqrt{1 + m^2}} \quad (4.24)$$

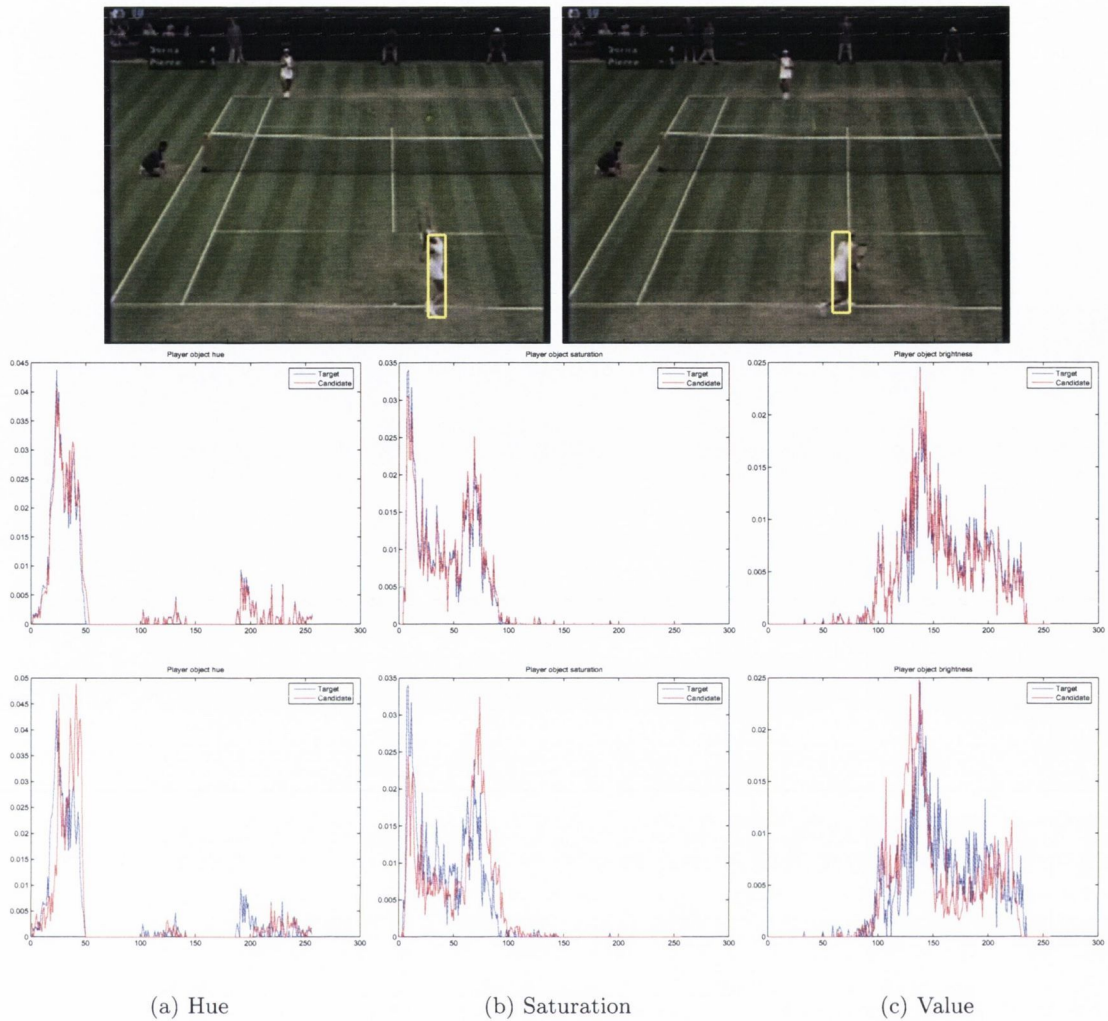


Figure 4.12: Bottom player location detected. Top: Player location at frame t and $t + 50$. Middle: Candidate and target histograms at frame t . Bottom: Candidate and target histograms at frame $t + 50$.

4.6.2 Angle between least squares fit and true trajectory

The second measure used is the angle between the true trajectory and the least squares fit to the data. The ground truth was found by manually locating the start and end locations of the ball across the trajectory. The least squares line was found using equation 4.25 where T is the number of frames in the shot.

$$\begin{bmatrix} m \\ c \end{bmatrix} = \begin{bmatrix} T & \sum_{i=1}^T x_i \\ \sum_{i=1}^T x_i & \sum_{i=1}^T x_i^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=1}^T y_i \\ \sum_{i=1}^T x_i y_i \end{bmatrix} \quad (4.25)$$

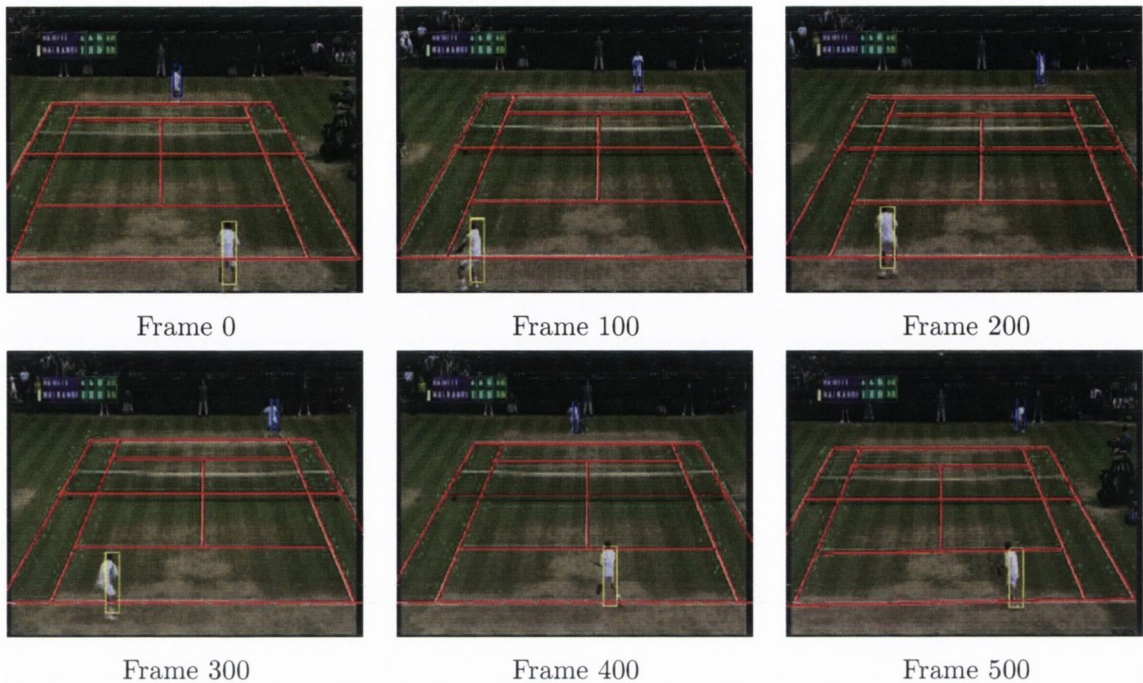


Figure 4.13: Player tracking of a rally event. Both players are tracked over a sequence of 500 frames and an overlay of the court is also provided.

The angle between the true trajectory and the least squares fit can then be calculated using the trigonometric expression below, where m_1, m_2 are the slopes of the two trajectories. This can be considered as being a measurement of deviation between the two trajectories.

$$\theta = \arctan \frac{m_1 - m_2}{(1 + m_1 m_2)} \quad (4.26)$$

4.6.3 Comments on the tracking performance

Results of the performance in terms of the mean distances from the true trajectory to the points along the track produced by the particle filter are tabulated in table 4.1. The mean distance for all tracks is accurate to a sub-pixel level, which is acceptable for this application. Furthermore, the mean angle difference between the true trajectory and the least squares fit also achieves sub pixel accuracy. Results are tabulated in table 4.2.

On tests of ball tracking, the white ball was successfully tracked 100% of the time using likelihood ratios and 90.9% of the time using likelihoods based on a ball colour model alone. There are some occasions when the tracker cannot successfully track some of the coloured balls. Correct tracking of balls in complex shots is intractable. For example, when a head-on collision⁵ between two balls of the same colour occurs, the ball in advance will be tracked.

⁵A head-on collision is also known as a 'flush' collision.

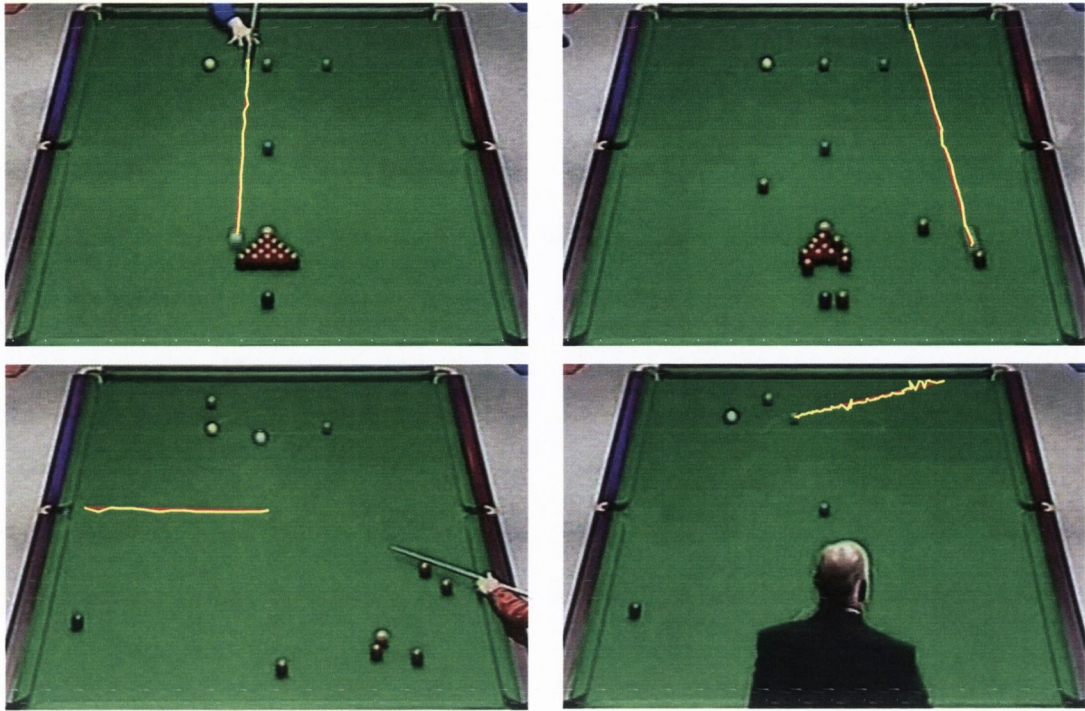


Figure 4.14: *Illustration of the performance of the particle filter for tracking: The tracks in red are the true trajectories of the ball while the yellow tracks are the particle filter estimation to the trajectory. Clockwise from the top left, examples of the individual tracks of two whites, the green and blue are shown. Note in particular the successful track of the green ball (bottom right) which is difficult given the similarity between the ball and table cloth colours.*

This can be remedied by applying the algorithm outlined in section 5.2 to detect collisions between all balls. When a collision is detected, the motion model directing the placement of particles can be reset to account for the sudden reduction in velocity of the current ball being tracked.

Motion blur is the main contributing factor leading to poor tracks in snooker. As the ball becomes elongated, the tracker will lock on to an area within this region which may not be the actual ball centre. The use of likelihood ratios remedies this somewhat but occasionally the hypothesised location of the ball will not always be located on the true centre.

If a ball is near a pocket and the player walks in front of the camera blocking the ball from view, a pot will be detected. This means that the tracker cannot always be fully relied on for detecting successful ball pots. The algorithm used in Denman et al [40], could be used to ensure that all the pots are detected and therefore if there is ambiguity in the semantics, they

Colour	White	Red	Black	Blue	Brown	Green	Pink
Shot 1	0.523	1.006	-	-	-	-	-
Shot 2	1.069	0.523	-	-	-	-	-
Shot 3	0.645	1.414	-	-	-	-	-
Shot 4	0.778	0.625	-	-	-	-	-
Shot 5	0.884	-	0.771	-	-	-	-
Shot 6	1.013	-	1.094	-	-	-	-
Shot 7	0.511	-	-	0.763	-	-	-
Shot 8	0.705	-	-	0.813	-	-	-
Shot 9	0.484	-	-	-	0.785	-	-
Shot 10	0.97	-	-	-	-	1.517	-
Shot 11	0.862	-	-	-	-	-	0.917

Table 4.1: Mean distances from the true trajectory to the projected points along the track produced by the particle filter.

Colour	White	Red	Black	Blue	Brown	Green	Pink
Shot 1	0.15	0.503	-	-	-	-	-
Shot 2	0.411	-0.676	-	-	-	-	-
Shot 3	0.154	-0.648	-	-	-	-	-
Shot 4	-0.054	0.981	-	-	-	-	-
Shot 5	-0.089	-	0.487	-	-	-	-
Shot 6	0.874	-	-0.345	-	-	-	-
Shot 7	-1.008	-	-	0.614	-	-	-
Shot 8	0.954	-	-	0.748	-	-	-
Shot 9	0.638	-	-	-	0.974	-	-
Shot 10	0.095	-	-	-	-	-1.115	-
Shot 11	-0.71	-	-	-	-	-	0.84

Table 4.2: Mean angles between the least squared trajectory and the true trajectory in $^{\circ}$.

could be verified. The representation of the game semantics will be presented in Chapter 6.

Tracking the tennis player proved to be more difficult. Histograms by their nature incorporate no spatial information and for a large irregular shape such as the tennis player, a correct lock on the centre of the region can not always be presumed. Furthermore, owing to the shape of the tennis player, the rectangular region not only contains player pixels but also

includes several pixels of the court. While they can be slightly suppressed using the weighting kernel, they do affect the tracking fidelity.

Another problem encountered was that of the occasional disappearance of the player from view when the camera pans to follow the trajectory of the ball. The track of the player was recovered by assuming that he will reappear close to the position from which he disappeared. Tracking of players in a doubles match was not considered for this work but may complicate matters with frequent occlusions.

As is evident from figure 4.14, the tracks for some of the balls are ‘noisy’. The apparent noise is due to the non-fractional accuracy of the random tracker. This is particularly evident in the track of the green ball. Due to the similarity between its colour distribution and that of the background, tracking becomes difficult. Also contributing to the noise is the relatively slow speed at which it is travelling. The track obtained is therefore perceived as not being as smooth as for the ball which move at a faster speed.

4.7 Tracking comparison

As a comparison of tracking performance, an implementation of the gradient based motion estimator in [81] was used to track snooker balls. The idea was to generate motion vectors for each pixel in the image (optic flow). The concatenating vectors in time, starting with an object position, yields an object track. The motion estimation methods enable those tracks generated by the particle filter to be evaluated and justify its use for tracking. This idea does not seem to have been considered in the literature.

4.7.1 Gradient Based Motion Estimation

A gradient based approach to motion estimation (GBME) involves expanding the generalised spatio-temporal model for motion in image sequences is given by equation 4.27.

$$I_t(\mathbf{x}) = I_{t-1}(\mathbf{x} + \mathbf{d}_{t,t-1}) \quad (4.27)$$

This equation describes how the image at time $t - 1$ can be mapped to that at time t by accounting for the displacement of magnitude $\mathbf{d}_{t,t-1}$ that the image undergoes. \mathbf{d} is known as the motion vector. A Taylor Series expansion of equation 4.27 yields the expression

$$I_t(\mathbf{x}) = I_{t-1}(\mathbf{x}) + \mathbf{d}^T \nabla I_{t-1}(\mathbf{x}) + e_{t-1}(\mathbf{x}) \quad (4.28)$$

The block matching solution to motion estimation defines the displaced frame difference (DFD) as:

$$\text{DFD}(\mathbf{x}, \mathbf{d}) = I_t(\mathbf{x}) - I_{t-1}(\mathbf{x} + \mathbf{d}) \quad (4.29)$$

Making use of the DFD, a solution for the motion vector *for each block* can be obtained by neglecting the higher order terms of the expansion $e_{t-1}(\mathbf{x})$ and vectorising equation 4.28.

$$\mathbf{d} = [\mathbf{G}^T \mathbf{G}]^{-1} \mathbf{G}^T \mathbf{z}_o \quad (4.30)$$

Where \mathbf{G} is a vector of intensity gradients at time $t - 1$, \mathbf{z}_o are the corresponding DFDs, obtained from equation 4.29 and \mathbf{d} is the motion vector.

This method is explained in greater detail in [81].

4.7.2 Quantitative comparison of tracking performance - GBME vs PF ⁶

The implementation of the motion estimator uses 9x9 blocks. Given that the radius of the ball varies from approximately 5 to 7 pixels, and even greater with motion blur, the accuracy of the GBME for ball tracking should be reasonably good. An illustration of the performance differences between the particle filter and the tracking using motion estimation is shown in figure 4.16.

Only the white ball is tracked in these frames because the GBME implementation does not allow differentiation between ball colours. Only one moving object can therefore be tracked at one time, so all other balls must be stationary. The same method employed in section 4.4.1 is used to find the frame in which the white ball begins its motion. In each frame, the weighted mean of the vectors in a 45×45 pixel region around the strongest vector on the table, is taken to be the position of the cue ball. To ensure that only motion due to the white ball is accounted for the analysis, any motion by the player must be removed. This is achieved by masking him from view using the method outlined in section 4.4.1.

Figure 4.15 shows the motion vector field for three frames from two shots used for measuring the performance of block matching tracker. Also shown is the track borne out by the white ball over the duration of the entire shot using the weighted mean of the vectors as the location. Figure 4.16 illustrates a comparison of the tracking achieved using both methods.

Tracking using motion estimation was assessed using the same geometrical metrics used in section 4.6. The true trajectory was obtained by manually locating the centre of the white ball from the first and last frames of the footage. Analysis was carried out on 12 tracks of the white ball and the results of the tracking are compared to those obtained using the particle filter. The shots varied in duration, the shortest of which is six frames and the longest, thirty. Table 4.3 compares the results of two tracking methods and shows the particle filter to be better in all cases.

4.8 Summary

This chapter presented a probabilistic colour based object tracker derived from the CONDENSATION algorithm and considered its use for tracking of objects in broadcast sports footage. Novel extensions to the tracker were conceived which make use of prior scene geometry and the known background colour distribution. This was shown to improve tracking

⁶The ME considered was realised in C, and took approximately 2.5 seconds per frame. The particle filter was implemented in Matlab and took approximately 10 seconds per frame depending on when convergence was reached and the number of particles used to estimate the location (100 for tracking snooker balls).

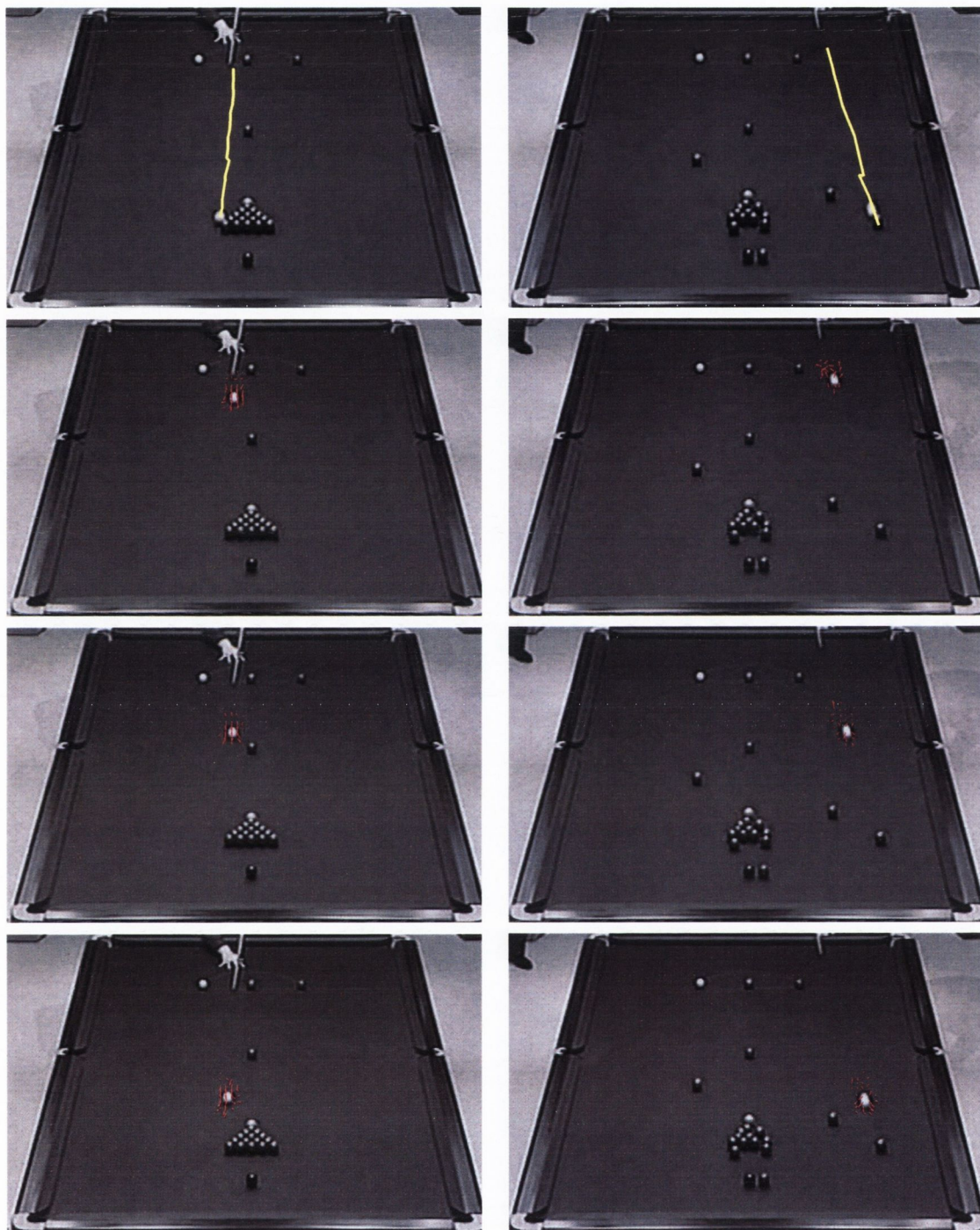


Figure 4.15: Motion vector field for two snooker shots. The full track of the ball using the weighted mean location is shown in the first row.

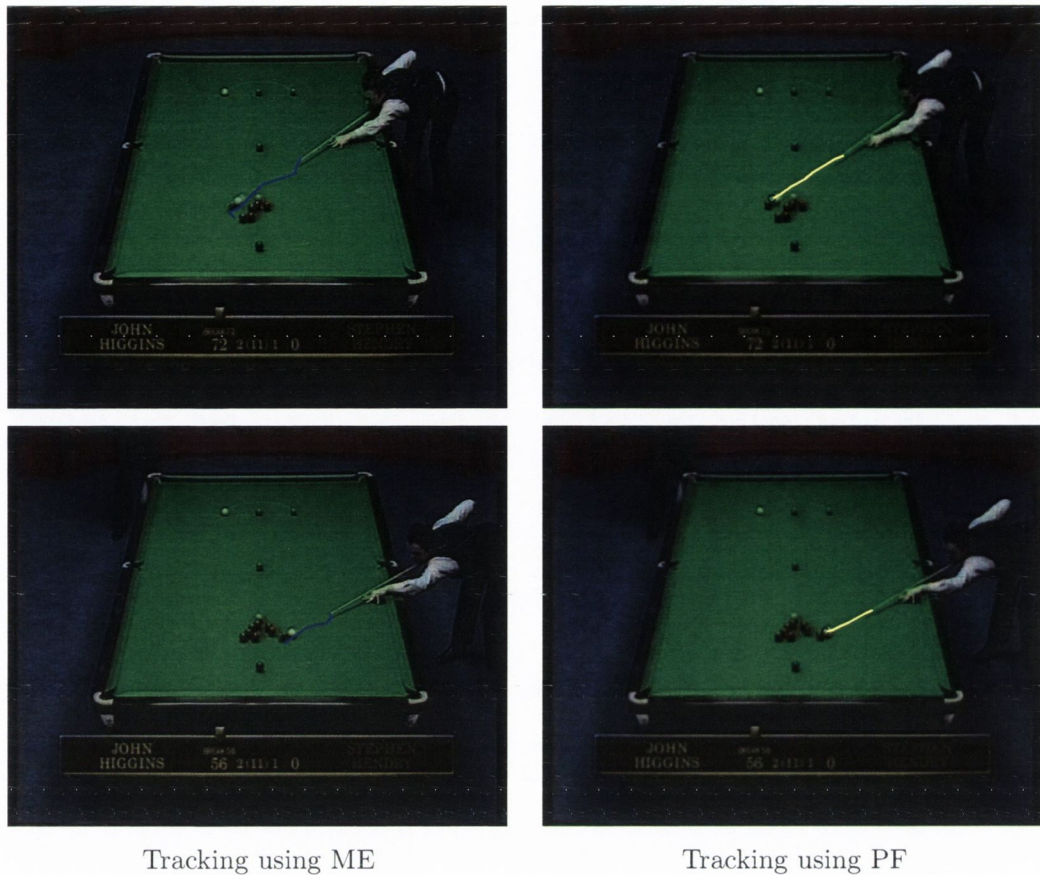


Figure 4.16: Comparison of tracking using gradient based motion estimation (left) and particle filtering (right).

fidelity and was used to good effect for tracking snooker balls and tennis players.

The performance of the tracker was assessed using geometrical measures and compared to the results obtained using a gradient based motion estimator. In chapter 6 the tracks provided by this chapter, in conjunction with results from chapter 5, will be exploited to retrieve high-level events which occur in tennis and snooker games.

Metric	Mean Distance PF	Mean Angle PF	Mean Distance ME	Mean Angle ME
Shot 1	0.523	0.15	1.201	2.093
Shot 2	0.754	0.744	0.950	-2.683
Shot 3	1.069	0.411	1.479	-1.522
Shot 4	0.979	-0.544	0.967	-3.527
Shot 5	0.845	-0.345	2.493	-1.258
Shot 6	0.595	0.078	3.376	1.107
Shot 7	0.645	0.154	2.753	2.167
Shot 8	0.837	-0.132	1.072	2.802
Shot 9	0.778	-0.054	2.336	1.541
Shot 10	0.690	0.048	1.002	-1.221
Shot 11	0.884	0.089	0.952	3.665
Shot 12	0.974	0.103	1.447	-1.784

Table 4.3: Table illustrating the performance difference between PF and ME for tracking of the white ball in terms of the distance and angle measures. Distances are in pixels and angles are in $^{\circ}$.

5

Dynamic Event Detection in Snooker

Snooker requires the player to accumulate the highest score possible by potting the coloured balls in a certain sequence (see appendix B for a brief description of the rules of the game and means for accumulating a high score). This can only be achieved by hitting the white ball and causing a collision with a particular coloured ball resulting in a pot. If a coloured ball is not hit or if an incorrect colour in the sequence is hit, a foul is called. Therefore, a semantic episode is expressed between the instant a player initially hits the white ball and the time at which all balls being tracked come to rest or are potted.

Within this period several incidents may occur which will affect the viewer interpretation of the shot made by the player. Incidents such as inter-ball collisions, ball-cushion bounces and ball pots, are events which determine this interpretation. These events can be inferred from the explicit tracking path approximated by the particle filter described in the last chapter. This is the focus of dynamic event detection.

Given that the ball spot positions are known at the start of the game, colour models for each ball can be established. This allows the colour of the ball which has just undergone collision to be detected. This is described in section 5.1. Section 5.2 outlines a method centred around segmentation and transient motion differences to detect inter-ball collisions and ball-cushion bounces. In section 5.3 a technique that utilises the sample likelihoods generated by the particle filter for the detection of ball pots is discussed. Furthermore, a foul can be inferred by incorporating collision detection techniques and analysing the trajectory of the white ball.

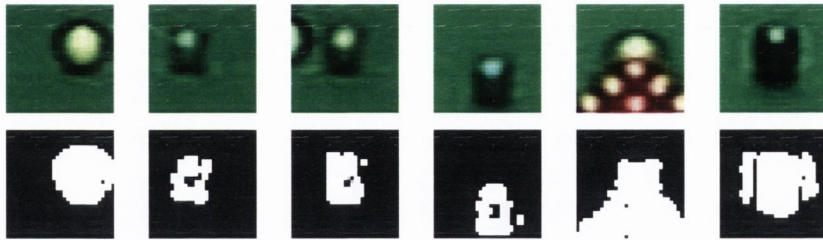


Figure 5.1: Segmented balls. Left to right: Yellow, green, brown (and part of the white ball), blue, pink, black and red.

5.1 Establishing initial ball colour models

Given that the position of each ball can be retrieved using the table finding technique outlined in section 3.3.2, colour models for each ball on the table can be generated automatically. This enables the colour of the ball which has just been hit to be determined (see section 5.2 for details on how to detect the collision). Detecting this colour provides a simple, but important piece of information which could prove useful to the user (for browsing) and broadcaster (for applications such as automatic scoring, *etc.*).

We assume that in the first frame of the global view, a new game has just begun and the balls are all on their appropriate spots. A small region, W , of 30×30 pixels centred on the estimated spot position of each coloured ball is then segmented. The problem of segmenting these balls poses a similar problem to that of the playing area segmentation discussed in chapter 3. Direct thresholding, adaptive thresholding and a GMM were once again employed for segmentation. The poor quality of the footage inhibited the performance of the GMM. Ghosting artefacts around the balls resulted in excessively large areas being detected and often caused objects which are close, to be merged. Direct thresholding of the difference of RGB colour planes proved equally inept over all footage sources. The adaptive thresholding method implemented here firstly locates a flat area of table, \mathcal{R} (in the same way as generating the texture for the player mask outlined in section 4.4.2). The luminance component, Y , of window W , is segmented using a twin density slice of the luminance component, where the thresholds are based on the local luminance statistics of the region \mathcal{R} (equation 5.1). $\sigma_{\mathcal{R}}$ is the standard deviation over the region \mathcal{R} and $i, j \in W$.

$$c(i, j) = ((Y_W(i, j) > \max(Y_{\mathcal{R}}) + 2\sigma_{\mathcal{R}}) \wedge (Y_W(i, j) < \min(Y_{\mathcal{R}}) - 2\sigma_{\mathcal{R}})) \quad (5.1)$$

Figure 5.1 shows the segmentation of each ball. Since the radius of the ball at each spot position is known, the colour information can be extracted from the circular region centred on the segmented object.

5.2 Collision Detection

Consider that the white ball has been hit. A straightforward approach to collision detection is to observe the change in velocity of that ball as it traverses the table. Abrupt velocity change indicates collisions. This information can be extracted directly from the particle filter tracker presented in the previous chapter. In practice however, the resolution of the tracker is limited by the resolution of the standard definition TV image. Here, slight collisions may not result in observable velocity changes. Fortunately frame differencing in the region of the impacting ball (white in this case) is always able to highlight another object responding to an impact. The basic idea proposed for collision detection is therefore to threshold the frame difference within a window W , the size of which is conditioned by the maximum velocity of the ball.

A window of twice the size of the maximum speed of the white ball is used to frame the impact area. The position of the centre of the window depends on the dominant velocity of the white ball. For example, if the magnitude of the horizontal velocity in the left direction is greater than its vertical velocity, the window is offset such that the distance from the white ball to the right hand side of the window is $\frac{W}{4}$. The same approach of adjusting the position of the window is adopted for all other dominant directions of velocity. This ensures that white ball is close to the edge of the window, giving a better chance of finding the ball which has just undergone impact. An illustration of the windowing is shown in figure 5.2.

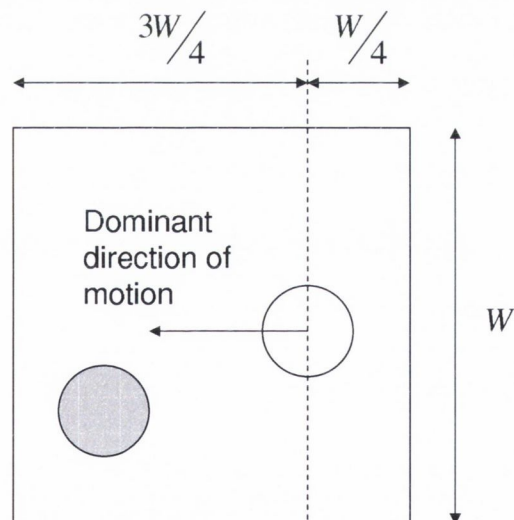


Figure 5.2: Impact area windowing.

Given the position of the white ball and its corresponding radius, the cue ball can be masked from view in each frame with suitable table texture. The same texture as exploited for player masking (section 4.4.1) can be used for this purpose. Any motion in the windowed

region must therefore only be attributed to the coloured ball which has just undergone the impact. If the inter frame difference in the window region W of the luminance component Y , between times t and $t - 1$ exceeds a threshold, a collision (*i.e.* coloured ball motion) is assumed to have occurred (equation 5.2).

$$\left(|Y_W^{(t)} - Y_W^{(t-1)}| \right) > 20 \quad (5.2)$$

5.2.1 Dealing with shape distortion

Upon detection of the collision event, the shape of the new ball may be distorted by motion blur. As a result, the modelled histogram will be corrupted by motion artefacts. As is the case for localising the white ball, a frame from the footage must be chosen such that the coloured ball to be tracked is motionless. To find the frame in which this ball is stationary, previous frames in the regions of impact are retroactively searched for lack of motion. This is achieved by using the same frame differencing method for detecting the collision. Once the non-motion frame has been found, the colour ball needs to be segmented from the background. This segmentation is attained by employing the same adaptive thresholding method outlined in section 5.1. The segmentation of several balls from the different footage sources is shown in the middle column of figures 5.3-5.4.

The centre of a connected component region of area greater than $\frac{2\pi}{3}r^2$ pels (where r is the radius of an object in that area) is deemed to be the centre of the new ball to be tracked. The same procedure as that used for modelling the white (described in section 4.4.2) is followed for tracking the coloured balls. A number of examples of successful localisations of a second ball are shown in the right hand columns of figures 5.3-5.4.

Complications arise in selecting the ball to be tracked when it is located close to others within the window. In an attempt to correctly select the object which has just undergone an impact, it is necessary to reduce the potential ball misclassification by monitoring the motion of all objects in the windowed area. Connected component regions from the colour segmented binary map in which motion is first exhibited (*i.e.* the collision) are labelled, and the distance between the centre of each object and the white ball is computed. If the centre of the object of shortest distance to the white ball is less than $2r$ pixels, it is selected as the ball to be tracked and a model of the region is created. If this condition is not fulfilled, the distance is retroactively checked in 5 previous frames. If no object is found to be within this distance, tracking of the white resumes from the frame where it was thought a ball was hit.

Occasionally the ball to be tracked might be partially occluded by another ball of different colour. By only exploiting the binary map generated using the luminance segmentation, these separate regions become merged. Given that the colour models of all balls are known (from section 5.1) it was attempted to associate regions in the binary map with their corresponding colour model. The windowed *HSV* frames were masked using the luminance binary map so that only relevant ball regions were showing. A likelihood was computed on each of these

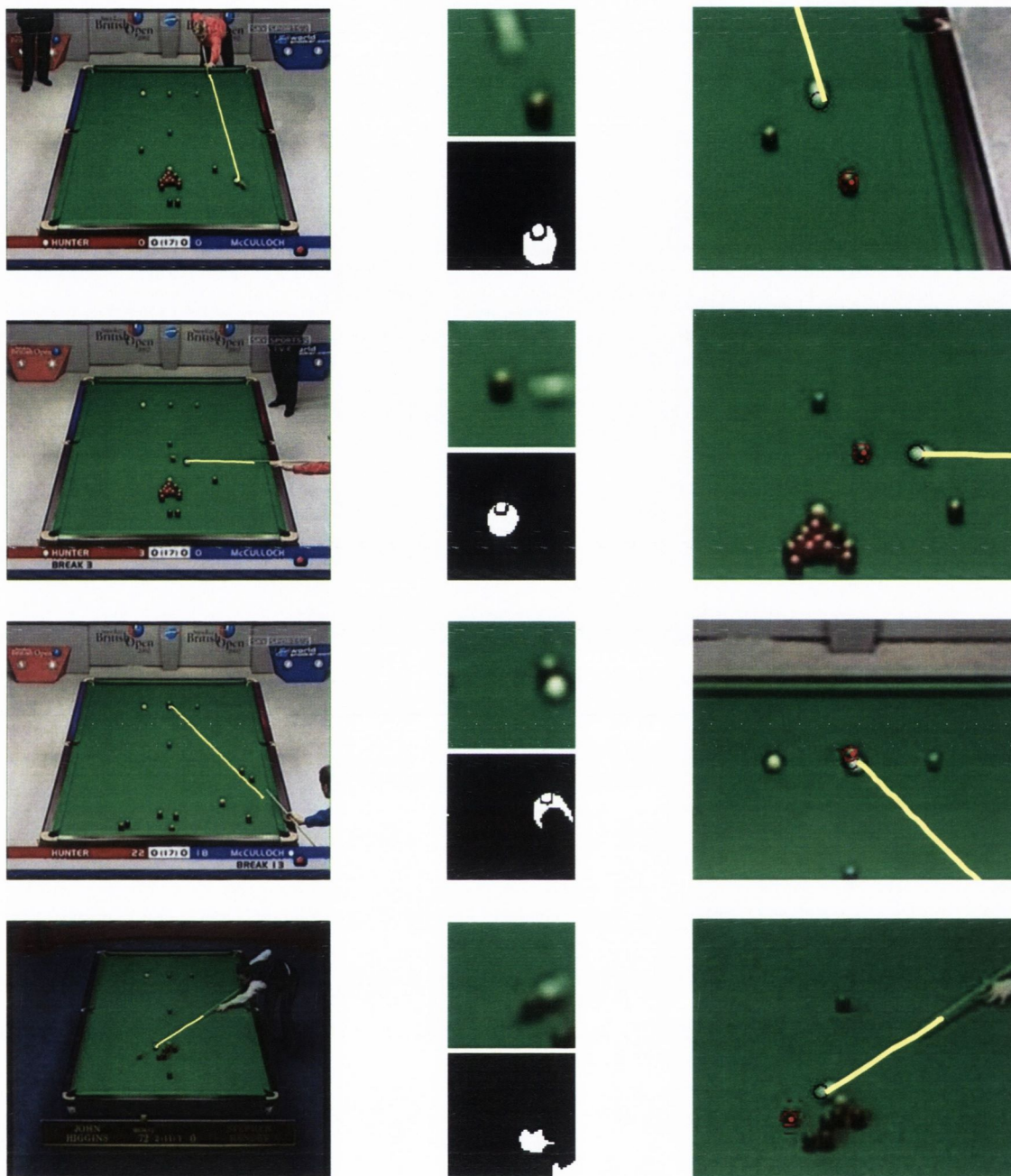


Figure 5.3: Left to right: Track of the white ball from its initial starting position until it collides with the coloured ball; (middle-top) The windowed balls before impact; (middle-bottom) A binary map of the above with the white ball masked; The coloured ball is found and colour properties are modelled.

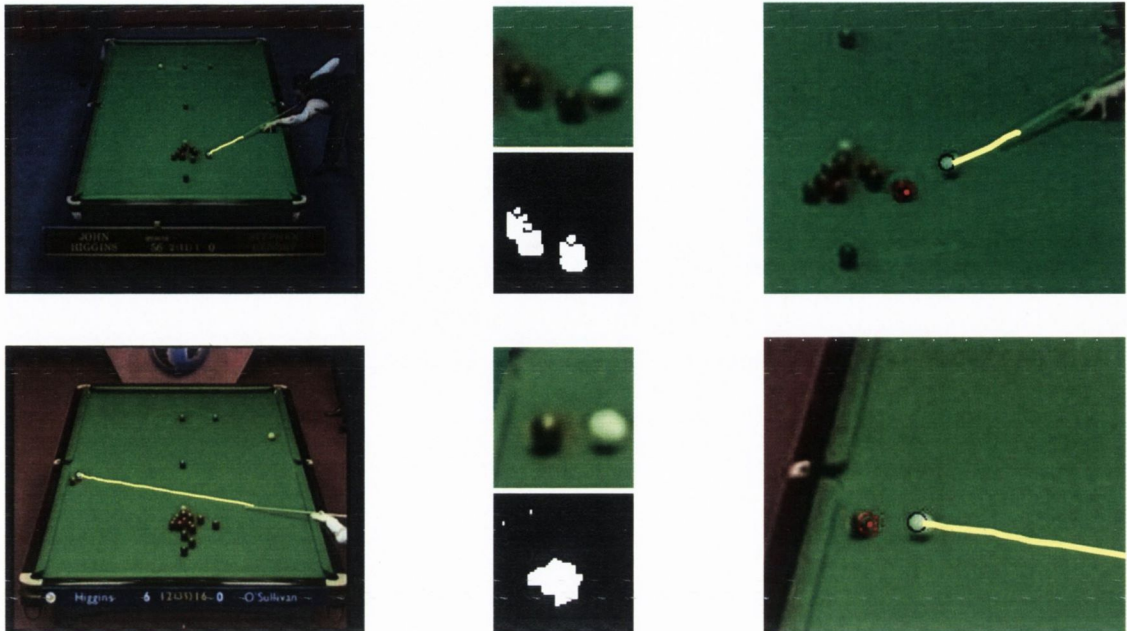


Figure 5.4: *Left to right: Track of the white ball from its initial starting position until it collides with the coloured ball; (middle-top) The windowed balls before impact; (middle-bottom) A binary map of the above with the white ball masked; The coloured ball is found and colour properties are modelled.*

pixels for each colour model. However, owing to the poor quality of the footage a reasonable segmentation was not achieved. Better segmentation methods and superior footage quality could resolve this issue.¹

5.2.2 Determining the colour of the new ball

To detect the ball colour, the likelihood of the colour distribution of the new ball region (obtained using collision detection) is computed, given the known ball colour models. The Bhattacharyya distance is once again used to calculate this similarity based on *HSV* histograms. Colour ball recognition could also be augmented by incorporating prior knowledge of the most likely location of the ball. For example, red and brown have similar colour distributions, however, red balls are more likely to be found in the bottom half of the table than the brown. Balls colours were identified with 86.67% precision.

¹The image on the left of the first row in figure 5.3 does not correspond to the detailed one in the middle because the white ball was travelling quite fast and due to the retroactive search outlined in section 5.2.1, the colour ball was found in the previous frame. The image on the right shows the ball located in the previous frame again.

5.3 Pot Detection

Monitoring the score in a snooker game aids in determining the state of the game. An event detection/object disappearance algorithm has been developed [40], which categorises the type of shot played as a miss, near-miss or pot. The classification is achieved by monitoring two rectangular regions centred on each of the pockets - a small one enclosed by a larger one. A ball pixel to table pixel ratio in the regions is calculated to detect the presence of incoming balls in each region. For example, if a ball enters the large region, then enters the small, leaves the small area first and then the large one, a miss can be inferred.

In this work, the particle filter can be used to achieve the same goal. When a ball is potted, it is obvious that the particle filter will be unable to continue tracking. The evidence for this drastic loss of lock can be extracted by observing the weight of each particle. When the likelihood is low, a loss of lock is indicated, hence inferring a pot. A threshold of L_r on the sum of the likelihoods of all particles was used to achieve this. Note that the cumulative likelihood of the particle set is calculated as part of the tracking phase (section 4.4). In the implementation considered here, a 60% reduction in the cumulative likelihood between the current and previous sample set is taken to indicate that the ball has been potted (*i.e.* $L_t/L_{t-1} < 0.4$). Some examples of correctly labelled ball pots are illustrated in figure 5.5.

Unfortunately, hard inter-ball or cue-to-ball impacts can also yield a drastic likelihood drop. In order to differentiate between a pot and such impacts, it is assumed that the ball which has just undergone the collision cannot be potted in the first two frames following the impact. This allows the ratios of L_t/L_{t-1} to stabilise.

5.4 Foul detection

Fouls can be inferred by monitoring the ‘bouncing’ state of the white ball. The implementation of the inter-ball collision detector does not enable cushion-ball type bounces to be detected. They can however be detected by identifying changes in angle between two trajectories when the white ball is in the vicinity of a cushion (within 20 pixels). This is achieved much in the same way as the angles between the particle filter and true ball trajectories were calculated in section 4.6.2.

The trajectories \mathbf{l}_1 and \mathbf{l}_2 are defined for $t_0 < t - 4$ in equation 5.3 as:

$$\begin{aligned}\mathbf{l}_1 &= \mathbf{c}(t_0, \dots, t - 4) \\ \mathbf{l}_2 &= \mathbf{c}(t - 3, \dots, t)\end{aligned}\tag{5.3}$$

Where \mathbf{c} is the vector of positions of the white ball from frame t_0 until t . t_0 is initialised as the frame in which the white starts its motion. It is set to the current frame number, t , if a ball cushion bounce is detected. The least squares fit for \mathbf{l}_1 and \mathbf{l}_2 are calculated using equation 5.4 where $\{(x_i, y_i)\}$ are the set of ball positions for the relevant trajectories. The



Figure 5.5: Clockwise from top: Tracking a red ball struck and potted by the white into the bottom right pocket (a continuation of the footage in the top row of figure 5.3) - Shot to nothing. Tracking a red ball struck and potted by the white into the middle left pocket (a continuation of the footage in the second row of figure 5.3) - Break building. Tracking a blue ball struck and potted by the white into the middle right pocket - Break building.

angle between the two lines (with parameters $[m_{l_1}, c_{l_1}]$ and $[m_{l_2}, c_{l_2}]$), θ_t , at frame t is then computed using equation 5.5 where m_{l_1} and m_{l_2} are the slopes of the two trajectories.

$$\begin{bmatrix} m_{l_b} \\ c_{l_b} \end{bmatrix} = \begin{bmatrix} \tau_2 - \tau_1 & \sum_{i=\tau_1}^{\tau_2} x_i \\ \sum_{i=\tau_1}^{\tau_2} x_i & \sum_{i=\tau_1}^{\tau_2} x_i^2 \end{bmatrix}^{-1} \begin{bmatrix} \sum_{i=\tau_1}^{\tau_2} y_i \\ \sum_{i=\tau_1}^{\tau_2} x_i y_i \end{bmatrix} \quad \begin{cases} \tau_1 = t_0, \tau_2 = t - 4 & \text{for } b = 1. \\ \tau_1 = t - 3, \tau_2 = t & \text{for } b = 2. \end{cases} \quad (5.4)$$

$$\theta_t = \arctan \frac{m_{l_1} - m_{l_2}}{(1 + m_{l_1} m_{l_2})} \quad (5.5)$$

If the absolute difference in θ between frame t and $t - 1$ is greater than 10° (i.e. $\eta =$

$|\theta_t - \theta_{t-1}| \geq 10^\circ$) the ball is deemed to have deviated from its original path and a ball-cushion collision is inferred.

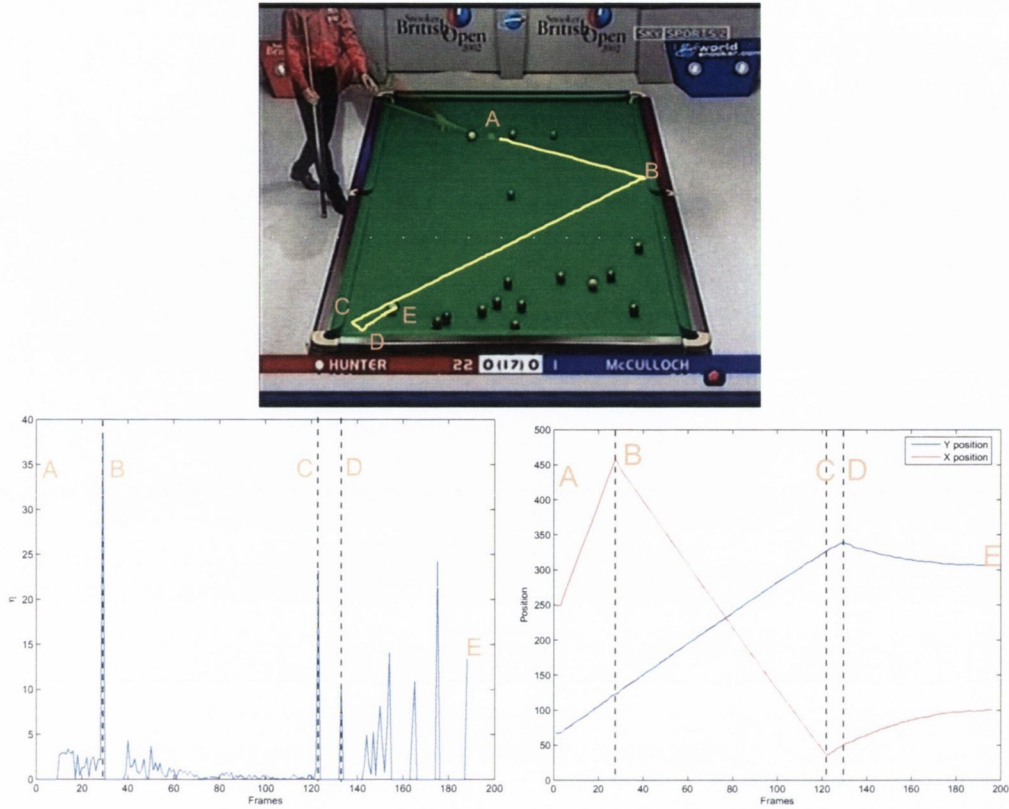


Figure 5.6: Foul detected due to the lack of inter-ball collision. Ball cushion bounces are detected at B, C and D in the plot. The impulses after D arise as the ball begin to slow down and so are not registered as cushion bounces.

Since both ball-cushion bounces and inter-ball collisions can be detected, a lack of ball-ball interaction before the white ball comes to a rest can be used to indicate that a foul has occurred. Figure 5.6 shows an example of such a foul event (top) along with a plot of the absolute angle derivative η (bottom left) and position of the white ball at each frame (bottom right).

As the snooker balls begin to slow down, the estimated locations of the balls are closer together. Under such circumstances, the derivative of the angle exhibits peaks in the plot (figure 5.6 (top) and 5.7 (top)). A condition on the speed of the ball (if the speed is greater than 3 pixels per frame) and the cushion location is therefore used to ensure correct ball-cushion collision detection.

5.5 Snooker escape

Detection of ball-cushion bounces also provides a clue to other semantic events. Assuming that a cushion bounce occurs before an inter-ball collision implies that the player is attempting a difficult shot. If the player approaches a shot in this way it can be assumed that the direct line of sight from the white ball to the ball which the player is attempting to hit is obstructed. This allows a snooker escape event to be inferred. A plot of the position of the ball for such an event is shown in figure 5.7 along with a track of the white ball and a plot of η . The cue ball is situated at the bottom cushion when the player begins his shot. He is attempting to hit the blue ball close to the top cushion.

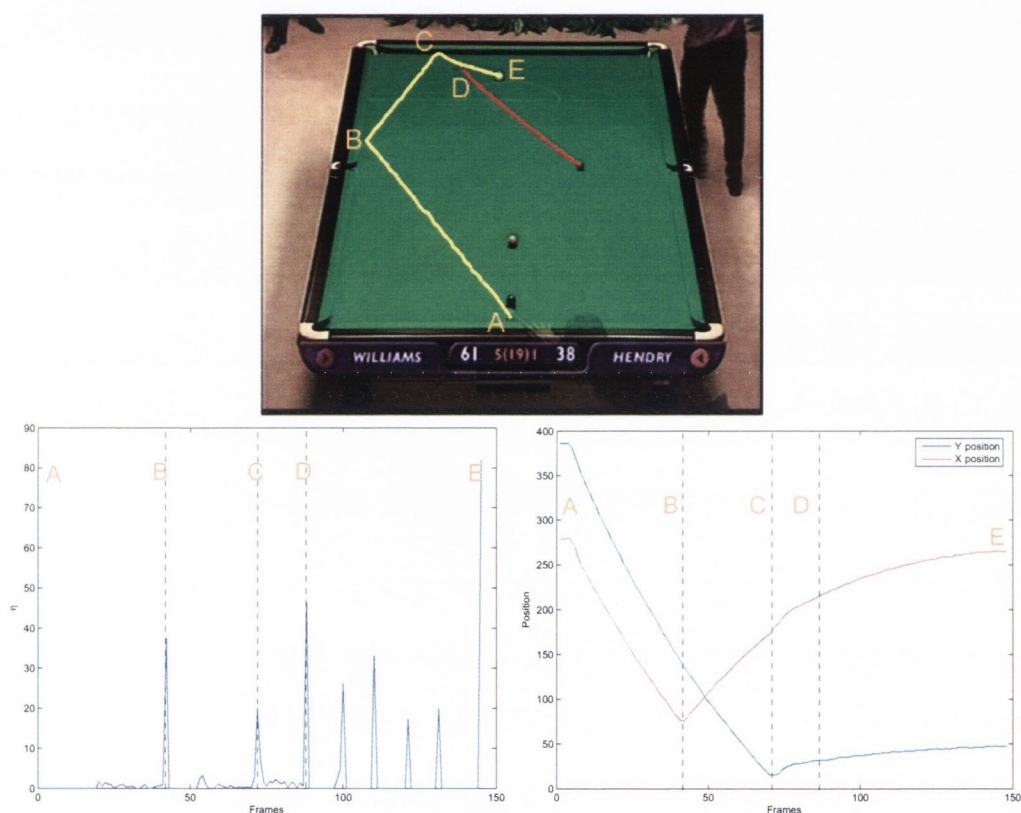


Figure 5.7: Example of a snooker escape. Two cushion bounces occur before an inter-ball collision. The bounces are clearly visible from the position plot while the plot of η validates the conditions for ball-cushion collision detection outlined in section 5.4.

5.6 Summary

In this chapter, the detection of incidents which affect the viewer's interpretation of particular events in snooker were discussed. These incidents included inter-ball collisions, ball-cushion bounces, ball pots and fouls. Results show the techniques used for the detection of collisions and pots to be robust for a number of different types of shot made by the players from a number of footage sources.

6

Event Modelling and Classification using HMMs ¹

In this chapter Hidden Markov Models (HMMs) are introduced as a means of modelling time varying patterns. Their use has found considerable success in applications where these patterns are particularly evident, for example in speech recognition [124, 128] and cognition based systems [140, 163]. Success in these fields has motivated their use in retrieval applications [5, 24, 76, 84].

HMMs are employed in two ways for modelling and classifying events that occur in snooker and tennis broadcast footage. Firstly, in order to conduct high level feature extraction such as ball tracking (section 4.4) and pot detection (section 5.3) in snooker and player tracking (section 4.5) in tennis, it is necessary to ensure that the correct view (*i.e.* that of the global view) is being shown. This can be achieved by modelling the stochastic structure of the moment features (the geometric Radon moment and statistical shape and colour moments presented in chapter 3) within each clip. A similar modelling technique was undertaken at the same time by Xie et al [160] for classifying ‘plays’ and ‘breaks’ in broadcast soccer footage.

As shown in figure 6.1, the evolution of the Radon moment feature is closely related to the view in each image. Similarly, the temporal behaviour of the shape and colour moments are correlated with the interleaving camera views. For these cases, the HMM can be used as a mechanism to link the behaviour of the moment features with the relevant camera views.

¹Results from this chapter have been published as “Modelling high level structure in sports with motion driven HMMs” by N. Rea, R. Dahyot and A. Kokaram in the *IEEE International Conference on Acoustics, Speech, and Signal Processing 2004* and in “Sport Video Shot Segmentation and Classification” by R. Dahyot, N. Rea and A. C. Kokaram in *Visual Communications and Image Processing 2003*.

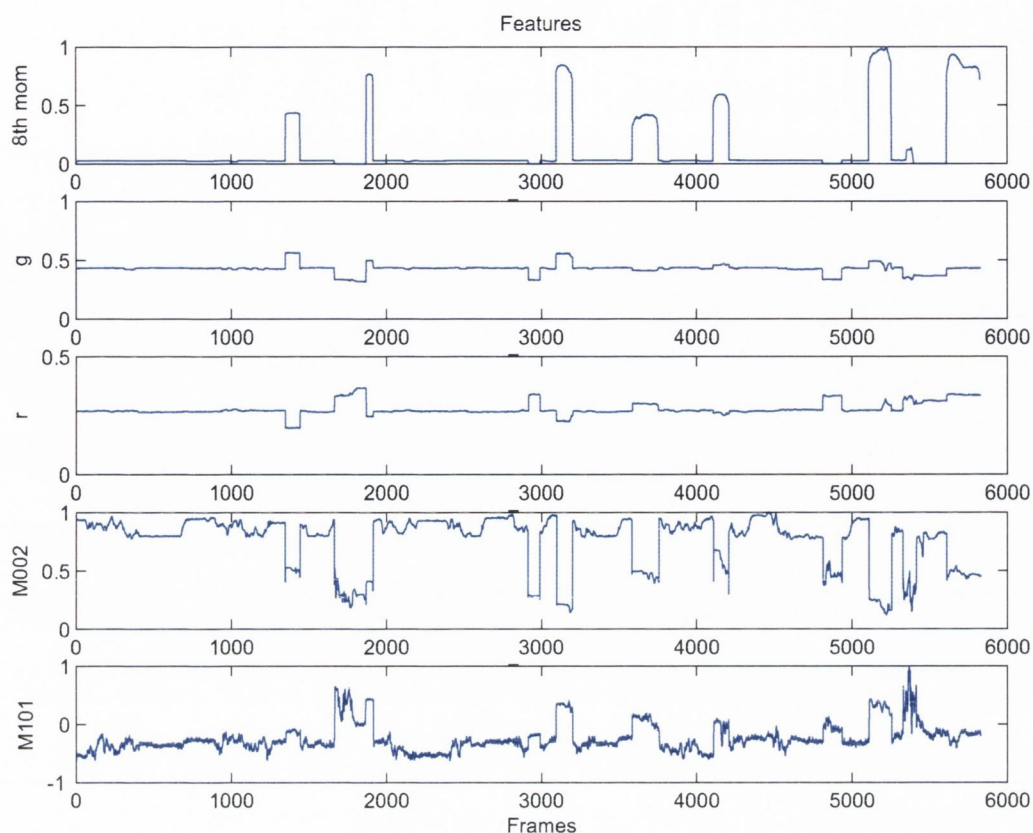


Figure 6.1: Moment features (from top) for snooker footage (Hendry): 8th order Radon moment; g chrominance; r chrominance; Second order moment of \mathcal{N} ; Second order moment of θ, \mathcal{N} .

Within each of the correctly classified global views, high-level feature extraction is conducted using a second HMM step. In this classification module, the spatio-temporal evolution of an object is considered to embody a particular semantic event. A HMM can be used to model the behaviour of this object as it traverses the playing area. In tennis we consider the movement of the player around the court to ‘mean something’, while in snooker the evolving position of the white ball over the duration of a shot can be related to a semantic event.

Discrete HMMs (DHMM) are used for modelling the temporal behaviour of both the moment features and the positions of the objects. This type of HMM is chosen over others (*e.g.* continuous HMMs and semi continuous HMMs) because the emission probabilities of the high level classification stage are a discrete distribution of quantised labels (*i.e.* quadrants of a playing surface). The moment features can be quantised in order to maintain the same HMM framework throughout.

To facilitate the generation of an alphabet to drive the DHMM, the moment features were quantised using a K-means clustering [45, 101] algorithm and a Gaussian mixture model (GMM) [102]. K-means clustering was used to generate a discrete alphabet of K entries representing the Radon moment feature, while the GMM was used to cluster the two dimensional statistical moment vector. For both cases, the choice of size of the codebook, or number of quantisation levels, results in a trade off between lower quantization error and faster HMM operations.

In all work using HMMs for retrieval thus far, there has been relatively little attention given to the details of HMM manipulation. The technique is certainly not well known in the image and video processing community while being fundamental to those groups in the speech processing. There are some important lessons to be learned in the workings of the HMM for video processing which will be presented in this chapter.

6.1 Creating the Alphabet

In order to drive the DHMM, the feature vectors must be quantised into a discrete set of labelled clusters. A cluster can be thought of as a set of points that are in some way related. Since it is common for data to exhibit similar properties, and therefore appear as clusters in feature space, algorithms have been developed which enable the understanding of the relationship that exist among the data.

Two common methods of clustering are K-means and Gaussian mixture modelling. The clustering transforms the continuous vector \underline{x}_n to scalar quantised levels X_t (alphabet). K-means is a simple, yet effective, optimisation algorithm which iteratively re-centres clusters by minimising a distance from all data points, $\{\underline{x}_n\}_{n=1}^N$, to the centroid of each cluster until convergence. The GMM involves parametrising the data set, $\{\underline{x}_n\}_{n=1}^N$, using a mixture of a predefined number of Gaussians. Means, covariances and weights of each mixture are iteratively estimated by the Expectation Maximisation (EM) algorithm. The GMM has previously been introduced in section 3.2.3 for an object segmentation application.

6.1.1 Clustering using the K-means algorithm

The K-means algorithm is known to be a good way of quantising one-dimensional real valued signals into a set of K discrete bins [45]. It is therefore appropriate as a clustering mechanism for the one-dimensional Radon moment. The algorithm works by firstly selecting (randomly or manually) K initial cluster centres from the data. K clusters are formed by associating each data point to the cluster centre to which it is closest. The centroids of the K clusters become the new cluster centres. The clusters are then individually labelled. These labelled clusters are known as the codebooks entries. For a large number of clusters the algorithm becomes computationally expensive being of complexity $\mathcal{O}(iKN)$ where K is the number of

clusters, i the number of iterations before convergence, N , the size of the data set.

The steps in the K-means clustering algorithm are outlined below.

1. Initialisation: Define the codebook size to be K and choose K initial cluster centroids. These centroids, μ_k for $1 \leq k \leq K$, are chosen at random from the existing data set of size N .
2. Classification: At the i^{th} iteration assign each data point $\{\underline{x}\}_{n=1}^N$ to a class, $C_k^{(i)}$ where $1 \leq k \leq K$, such that the distance of the data point to the centre of the class is minimised.

$$C_k^{(i)}(\underline{x}_n) = \arg \min_{1 \leq k \leq K} [\|\underline{x}_n - \mu_k\|] \quad (6.1)$$

3. Updating: Update each cluster $C_k^{(i)}$ by computing new cluster centroids $\hat{\mu}_k$ where $k = 1, \dots, K$. The cluster centroids are the mean values of all data points, \underline{x}_n for $1 \leq n \leq N_k$, associated with that cluster, where N_k is the number of training data in cluster $C_k^{(i)}$.

$$\hat{\mu}_k = \frac{1}{N_k} \sum_{x_n \in C_k^{(i)}} (\underline{x}_n) \quad (6.2)$$

4. Termination: If the decrease in the overall distortion, \mathcal{J} , at the current iteration compared with that of the previous one, is less than a particular threshold, then stop; otherwise goes back to step 2. The overall distortion is an objective function based on the distances between all data points in class $C_k^{(i)}$ and their associated cluster centre, defined in equation 6.3.

$$\mathcal{J} = \sum_{k=1}^K \sum_{x_n \in C_k^{(i)}} \|\underline{x}_n - \mu_k\| \quad (6.3)$$

Labelling the clusters

The erratic moment values that can be seen in the plot (figure 6.2) are due to global motion as the camera zooms and pans around the table. This resulted in a codebook length of 20 being used to cluster the Radon moment vector for *Higgins* and 24 for *Hendry*. The feature vector is transformed into the codeword which in turn is used to drive the HMM.

6.1.2 Clustering using Gaussian mixture models

The GMM has been introduced in chapter 3 where it was employed to model the colour distribution of a playing region for segmentation. In this section, the GMM is used for clustering of a feature space. It is clear from figure 6.4 that the clusters of feature points can not be modelled by a single Gaussian distribution. Therefore, the Gaussian mixture model implicitly assigns several Gaussians to model each such cluster 6.4. To reduce the chance of

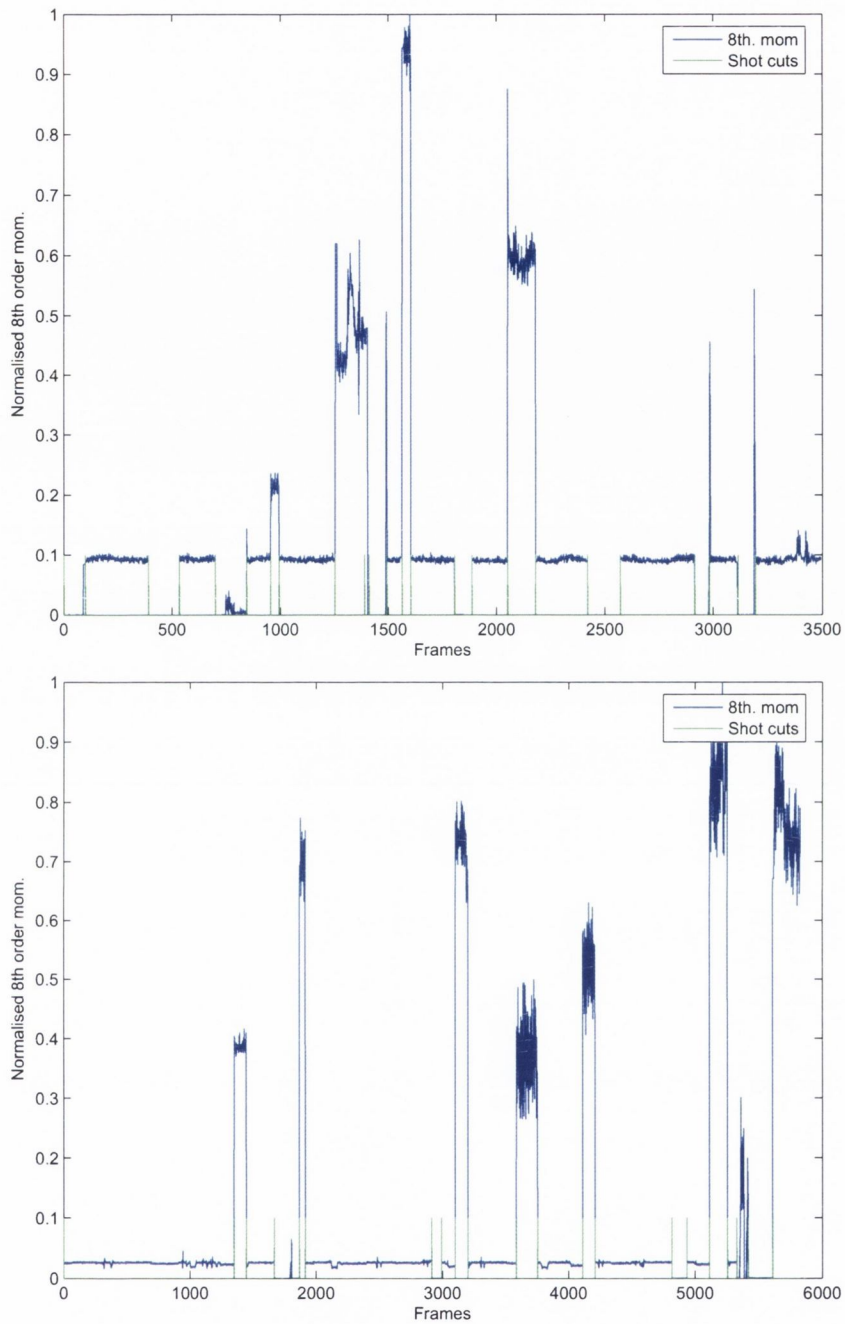


Figure 6.2: Shot cuts (green) are also shown along with the normalised 8th order Radon moment (blue) for Higgins and Hendry footage.

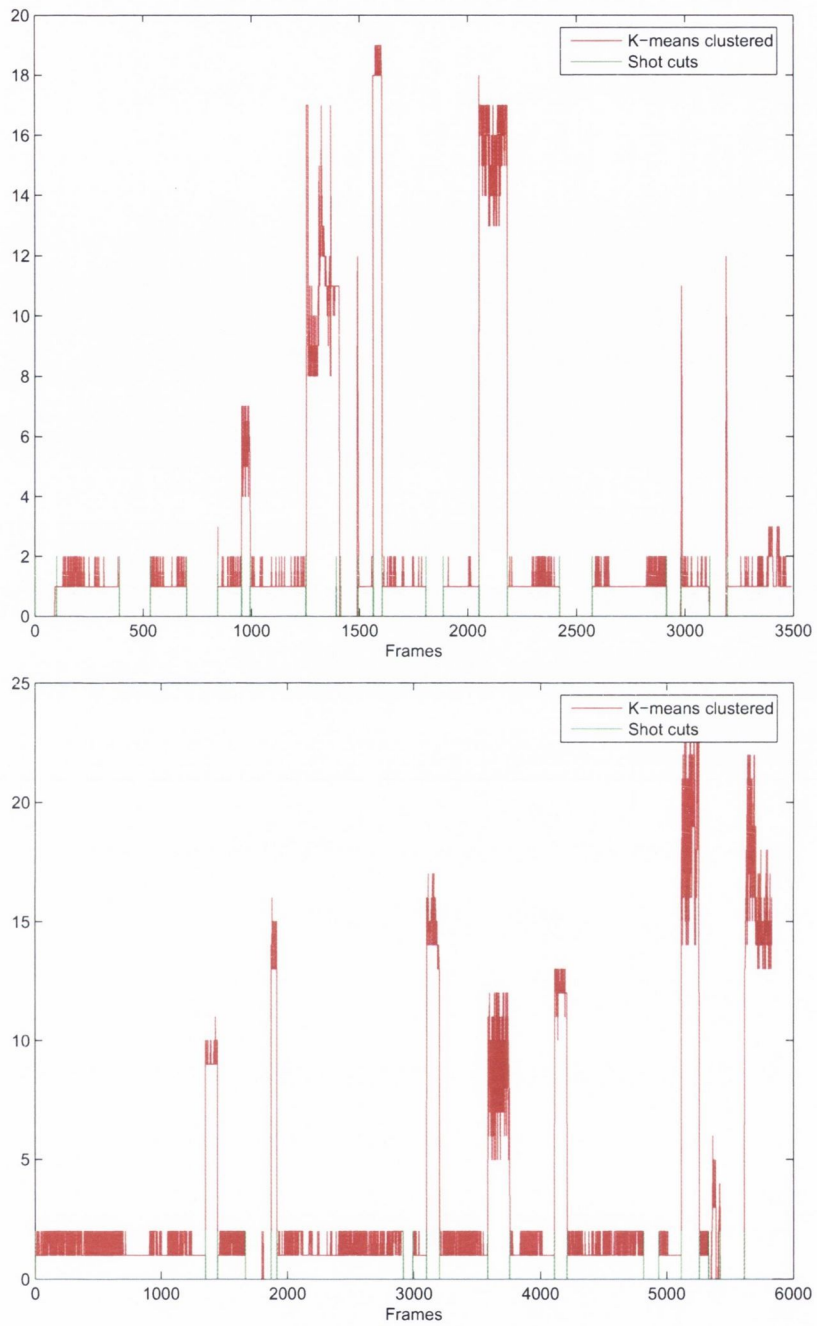


Figure 6.3: Shot cuts (green) and the clustered moment vector using K-means (blue) for Higgins and Hendry footage. The quantised levels are on the y-axis.

over fitting, the Gaussians estimated by the EM mixture modelling process are merged and pruned. On each iteration, all mixtures are tested for the following pruning and merging conditions:

- **Prune:**

If the probability of a particular Gaussian is 0. (*i.e.* $w_k = 0$).

If the $\det(\mathbf{R}) = 0$

If the condition number of the covariance matrix exceeds a specified threshold (10,000 in this case).

- **Merge:**

As the algorithm approaches convergence (a convergence tolerance of $2 \times tol$ is used, see section 3.2.3 for more details on convergence) the Euclidean distance between each cluster mean is calculated. If this distance is less than a specified threshold (0.01 for all footage), the cluster exhibiting the greatest weight (w_k) is kept and the remaining are eliminated.

Using equation 6.4, upon convergence each data point is labelled by associating it with the mixture component to which it is most likely to be a part.

$$C_k^{(i)}(\underline{x}_n) = \arg \max_{1 \leq n \leq N} [p(\underline{x}_n | C_k)], \quad 0 \leq k \leq K - 1 \quad (6.4)$$

For the colour moments for the snooker footage for example (figure 6.4 (left)), 25 initial mixtures were used to model the distribution. This was pruned to 22 mixtures and merged to 13. Each of these clusters is then assigned a discrete codebook entry for training and classification via the HMM.

6.2 Hidden Markov Models

A time domain process demonstrates a Markov property if the conditional probability density of current events, given all present and past events, depends only on the r^{th} most recent events. A r^{th} order Markov process is given in equation 6.5 where q_t is the hidden state of a system at time t .

$$p(q_{t+1} | q_t \dots q_1) = p(q_{t+1} | q_t \dots q_{t-r}) \quad (6.5)$$

This thesis concerns itself solely with first order models which are given by the expression 6.6.

$$p(q_{t+1} | q_t \dots q_1) = p(q_{t+1} | q_t) \quad (6.6)$$

This independence assumption also extends itself to model observations. If the t^{th} hidden state of a HMM is q_t and the observation generated from this hidden state is X_t , the

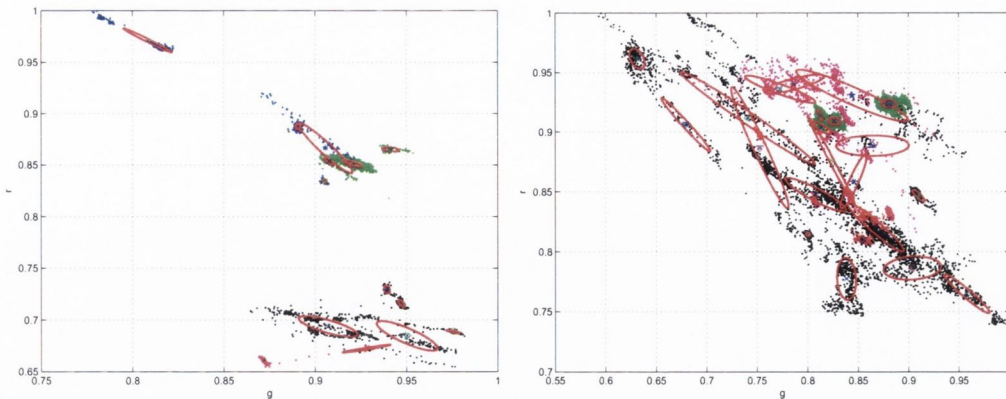


Figure 6.4: Colour moments distribution: Clustered moments using a GMM, 13 mixtures for Hendry (left) and 24 for Hewitt (right).

observation is independent of all other variables conditioned on q_t , where T is the length of the observation sequence $\mathbf{X} = (X_1 \dots X_T)$.

$$p(X_t | X_T \dots X_1, q_T \dots q_t \dots q_1) = p(X_t | q_t) \quad (6.7)$$

Unlike Markov models in which each state corresponds to an observable or physical event, HMMs include the case where observations are a function of the states. This means that a HMM can be implemented to represent the statistical nature of the observations in terms of a network of states. For each observation the process occupies a single particular state in the HMM. The current state is therefore conditioned only by the previous r states and the state transition probability associated with the state.

HMMs have found most use in problems which are inherently temporal. This temporality is particularly evident in visual and speech cognition based systems. The HMM approach to speech recognition was established by Lenny Baum in the early 1970's [105]. Since then there has been much investment in speech recognition systems such as "Natural Speaking" by Dragon² and Microsoft Speech³.

The earliest use of HMMs in the visual domain was by Yamoto et al [163] in which human actions were modelled and classified using a HMM. Visual recognition of sign language [140] and handwriting [164] have also been studied. The ability of the HMM to manage large deviations in feature behaviour such as those in these cognition based systems encourages its use for high-level video indexing problems.

There are two primary disadvantages of using HMMs. The first is the need for *a-priori* notation of the model topology. The structure of the HMM is data dependent and the obser-

²Natural Speaking: <http://www.scansoft.com/naturallyspeaking/preferred/>

³Microsoft Speech: <http://www.microsoft.com/speech/>

vations must be understood in order to create an efficient model. Secondly, large amounts of labelled training data are required to create a model that will work well. For this application, these disadvantages do not pose a problem. Prior knowledge of the sports allows the model topology to be easily created while video data for model training of the events are plentiful. Moreover, the observations used for high level event classification enable human specification of the training sequences. This is discussed in section 6.5.3.

6.3 Defining a HMM

A HMM can be defined by a set of five parameters. If the full parameter set is present the HMM can be used either as a generative model or to compute how likely a particular observation sequence is.

The elements of a general discrete HMM are:

- A set of states $S = \{S_1, S_2, \dots, S_N\}$. The state at time t is q_t . The process moves from one state to another in a Markovian fashion.
- Matrix of transition probabilities $\mathbf{A} = (a_{ij})$ where $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$, $1 \leq i, j \leq N$. It defines probabilistically how the process moves among states, where a_{ij} obeys the standard constraints:

$$\begin{aligned} a_{ij} &\geq 0, & 1 \leq i, j \leq N \\ \sum_{j=1}^N a_{ij} &= 1, & 1 \leq i \leq N \end{aligned} \quad (6.8)$$

A three state HMM with transition and observation probabilities is illustrated in figure 6.5.

- Set of discrete observations $V = \{x_1, x_2, \dots, x_K\}$: States are not directly observed. In a given state observations are generated according to a distribution described by \mathbf{B} , discussed next.
- Matrix of observation probabilities $\mathbf{B} = b_j(x_k)$ where $b_j(x_k) = P(X_t = x_k | q_t = S_j)$, $1 \leq j \leq N$, $1 \leq k \leq K$: It defines the pdf of observations given the state. In the discrete case \mathbf{B} obeys the constraints:

$$\begin{aligned} b_j(x_k) &\geq 0, & 1 \leq j \leq N \\ \sum_{k=1}^K b_j(x_k) &= 1, & 1 \leq j \leq N \end{aligned} \quad (6.9)$$

In essence, each state has a probability mass function associated with it that dictates how likely a particular observation is from that state.

- Vector of initial probabilities $\pi = \{\pi_i\}$; $\pi_i = P(q_1 = S_i)$, $1 \leq i \leq N$: It defines the probability of the initial state.

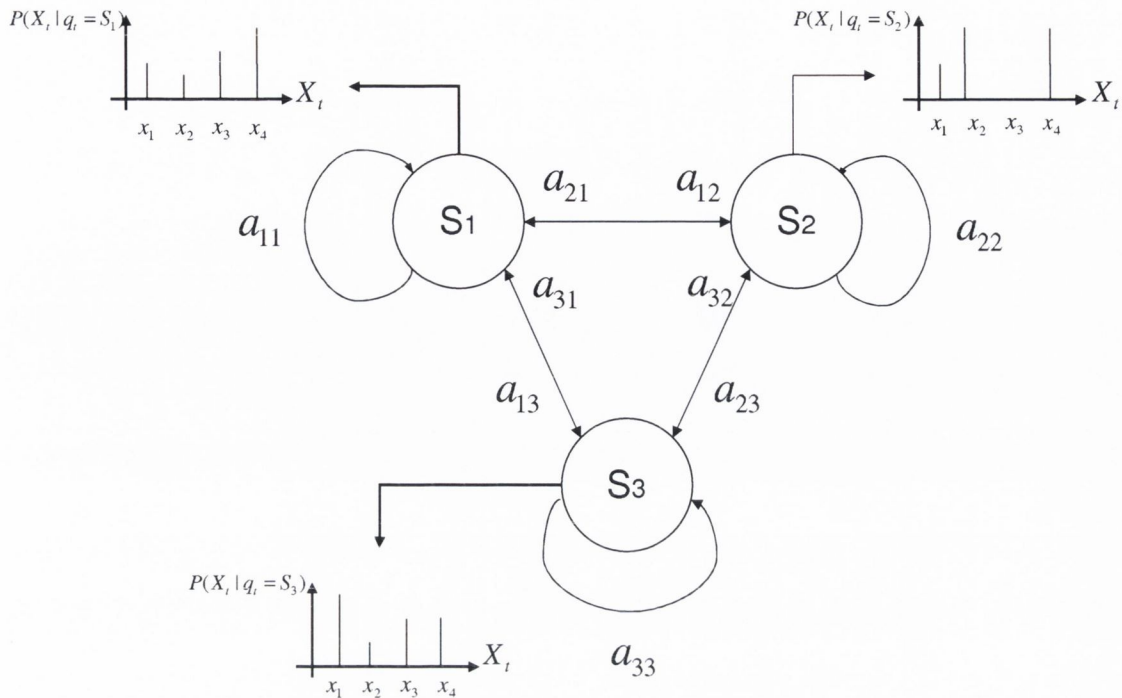


Figure 6.5: Three state ergodic HMM ($N = 3$) with transition probabilities, and four quantised levels in the codebook (discrete observations) $V = \{x_1, x_2, x_3, x_4\}$ (i.e. $K = 4$) with their probability mass function $p(X_t | q_t = S_i)$ where $1 \leq i \leq 3$.

So, given K and N a succinct definition of a HMM can be given by λ , where

$$\lambda = \{\mathbf{A}, \mathbf{B}, \pi\} \quad (6.10)$$

Three central issues of evaluation, decoding and estimation have to be resolved before a HMM can be applied to a specific problem.

Issue 1: Given the observation sequence of quantised levels $\mathbf{X} = \{X_1, X_2, \dots, X_T\}$, and the model $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, find the probability that the HMM actually generated the sequence regardless of the particular state sequence.

Issue 2: Given the observation sequence of quantised levels $\mathbf{X} = \{X_1, X_2, \dots, X_T\}$, and the model $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, find the most likely sequence of states, $\mathcal{S} = \{q_1, q_2, \dots, q_T\}$, that produced the observation sequence.

Issue 3: Given the observation sequence of quantised levels $\mathbf{X} = \{X_1, X_2, \dots, X_T\}$, calculate the model that best fits the data (i.e. evaluate the model, $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, that maximises the likelihood $P(\mathbf{X} | \lambda)$).

A comprehensive explanation of these three issues along with derivations of the appropriate parameters are given in appendix A.

For the two applications considered in this chapter (*i.e.* modelling the stochastic nature of the moment features and the spatio-temporal evolution of an object) only issues 1 and 3 are of importance. Issue 1 refers to the design of a sort of ranking system where there are a number of competing HMM classes and the one that best fits the observation is to be found. In other words, from an observation sequence, \mathbf{X} (which is either the quantised moment features or playing area quadrants), it enables classification based on maximising the expression in equation 6.11.

$$\hat{\lambda} = \arg \max_{1 \leq c \leq C} [P(\mathbf{X}|\lambda_c)] \quad (6.11)$$

For this work the resolution of issue 1 allows classification of camera view types and high level events (from C competing models $\{\lambda_c\}_{c=1\dots C}$). Issue 3 involves estimating the various model parameters from training data. An iterative EM algorithm called the Baum-Welch algorithm is typically employed. This algorithm is described at length in appendix A.

6.3.1 HMM topology

To determine the HMM topology is to choose the number of states and the connection between them (*e.g.* ergodic, left-to-right, *etc.*). In order to do so it is necessary to understand the meaning of the states. Various studies suggest that the HMM should be designed depending on the signal being modelled [124]. Since the observation vectors reflect the ‘real world’ representation of the hidden states it is necessary to choose the topology so as to reflect the stochastic nature of the observations.

Suggested methods for the choice of model have varied in the literature. Models based on intuition and empirical adjustment of the number of states has been advocated by Hu et al [65] where the representation of the states in real world terms is difficult to articulate. A data driven approach where the model is constructed from the data, reflecting the structure of the target pattern has been used by Lee et al [89]. In the design method of [89], the number of states in the HMM is determined by the structural decomposition of Korean characters into straight line segments. The direction of adjacent points on the line is clustered to form individual segments based on a set of 16 directional codes. A left-to-right topology was chosen to represent the temporal evolution of the line segments. In Kijak et al [76], each state in the HMM is chosen to represent a particular view or shot transition and the state transitions are derived from the observations and conventional editing techniques.

If the observation vector can be represented by a number of topologies, then the one of least complexity and fewest elements of model parameters is the best choice [124]. An excessively large HMM will incur increased computational cost while perhaps only marginally increasing recognition performance.

The number of states used in the event and view classification experiments attempts to conceptualise the nature of the data presented (section 6.5). This does not mean however that these models are the sole method of modelling the data. The essence of HMMs allow data to be modelled in various ways, none of which are *incorrect*. However, the same number of states and observations obviously need to be used in training as the detection.

6.4 View classification of snooker and tennis sequences

In order to conduct high level feature extraction such as ball tracking (section 4.3) and pot detection (section 5.3), it is necessary to ensure that the correct view (*i.e.* the global view) is being shown. While other views are not exploited for high level event detection in this research, they may find some use for extracting other useful semantics in future research such as in [5] and [76]. Parsing the footage using the low-level moment features described in chapter 3 is one way of accomplishing this. A HMM is constructed which is trained on half of the features and then tested on the entire sequence.

Xie et al [160] reduce soccer footage to the canonical forms of a *play* (*i.e.* when the ball is on the pitch) or a *break* (*i.e.* when the ball is over a touch line). The main aim of this research is not to locate semantic events in the footage, but to determine the *play* sections, which allows the footage to be condensed to less than 60% of the original length. The stochastic nature of two low-level features, dominant colour ratio and mean motion intensity, are modelled using HMMs. A Viterbi algorithm is then used to account for the long-term correlation of the play/break scenes. Results show classification accuracy of approximately 83.5%. This is similar to our method of classifying camera views which was published at the same time (section 6.4).

Sections 6.4.1 and 6.4.2 present the results of classification of shots using the Radon moment feature and a combination of shape and colour moments respectively. Table 6.1 indicates the length of the snooker and tennis sequences in terms of frames. The number of shots in the sequence is also listed along with the chosen number of classes of interest. An illustration of some frames from these classes is shown in figure 6.6.

As discussed in section 6.3.1, determining the number of states to model an observation sequence can be a difficult task and is generally empirically derived. As there exists no correct analytical means of calculating the optimum HMM topology for the purpose of the temporally evolving moment feature (*i.e.* associating states with observations), the most viable one is chosen empirically by determining that model which maximises the likelihood of each class while minimising computational effort. To this end, a two state ergodic topology (figure 6.7) is used to model the image sequences whose properties change over time. One possible way these states might be related to the observation sequence is that one state models the more homogeneous observation behaviours while the other state deals with the more impulsive observations.

Footage	<i>Clip1</i>	<i>Hendry</i>	<i>Hunter</i>	<i>Pierce</i>	<i>Malisse</i>	<i>Hewitt</i>	<i>Costa</i>
# Frames (Total)	3491	5832	24250	2949	4114	12009	11000
# Shots (Total)	23	21	115	16	18	59	75
# Global views	11	9	42	5	9	16	20
# Classes	4	5	5	3	4	4	4

Table 6.1: Snooker and tennis sequences. The classes for each footage source correspond to the images in figure 6.6 (e.g. the four view type classes in *Higgins* are the Global view, Playing area close-up, player and other area view).

6.4.1 View classification using the Radon moment feature

The HMM was employed to model the stochastic nature of the moment of the Radon transform. The models were trained using half the observations for each shot associated with the particular view type. This is illustrated using the first 2000 frames of quantised observations from *Higgins* in figure 6.8. The training vector is the hatched areas between the shot cuts shown in black.

Since the temporal boundaries can be detected using the methods outlined in chapter 3, the different shot types can be recognised in the context of the entire image sequence of an arbitrary length, by finding the most likely model according to equation 6.12. In this case, \mathbf{X} is the quantised Radon moment observations and \mathcal{C} is the number of views.

$$\hat{\lambda} = \max_{c=1 \dots \mathcal{C}} [P(\mathbf{X}|\lambda_c)] \quad (6.12)$$

The accuracy of the retrieval is given in terms of precision and recall which are defined below. To summarise from chapter 3, recall is a measurement of the ability of the HMM to retrieve all relevant views whereas precision measures the ability of the HMM to retrieve only relevant views.

$$Recall = \frac{A}{A+C} \quad Precision = \frac{A}{A+B} \quad (6.13)$$

A is the number of correctly retrieved views, B is the number of incorrectly retrieved views and C is the number of correct views which were missed by the classifier.

The main misclassification in snooker is in miscellaneous views of the crowd and score board, being classified as a close up of the player. This can be explained by observing the moment plots in figure 3.19, where the segmentation process results in a sparse Radon space. This is because no table colour, and hence geometry is found in these images.

Results of the classification using the Radon moment feature for tennis are not as good as for snooker. This is because of the wild deviations in behaviour of the features over shots which are similar (see figure 3.20). Global motion is the primary reason for this feature behaviour and hence low classification results.



Figure 6.6: Examples of frames from the classes of interest. Top to bottom, Higgins, Hendry, Hunter, Pierce, Hewitt1, Malisse, Costa.

	<i>Higgins</i>	<i>Hendry</i>	<i>Hunter</i>
Precision	95.65%	86.96%	82.86%
Recall	100%	100%	89.69%

Table 6.2: Precision and recall results for view classification using the Radon moment feature in snooker.

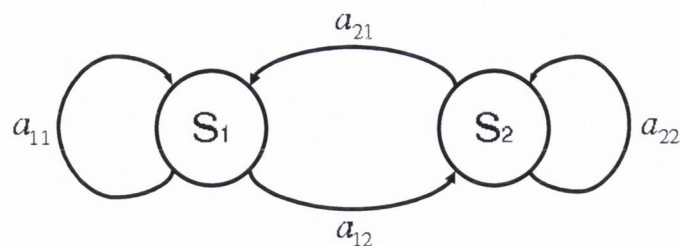


Figure 6.7: Two state ergodic model.

	<i>Pierce</i>	<i>Malisse</i>	<i>Hewitt</i>
Precision	52.24%	55.98%	57.26%
Recall	60.95%	67.64%	60.45%

Table 6.3: Precision and recall results for view classification using the Radon moment feature in tennis. *Costa* was not analysed for reasons outlined in chapter 3.

6.4.2 View classification using statistical colour and shape moments

In a similar fashion that of the previous section, the quantised colour and shape moments (generated in section 3.3.4) were used to drive a two state HMM. For this experiment, the HMM was used in an attempt to recognise views from three snooker sequences and four tennis sequence. The accuracy of the retrieval is once again given in terms of precision and recall.

The system was initially assessed using the colour and shape information individually. The classification results using colour alone were superior to the various combinations of coupled shape features.

The average performance of the classification achieved by each individual feature is given

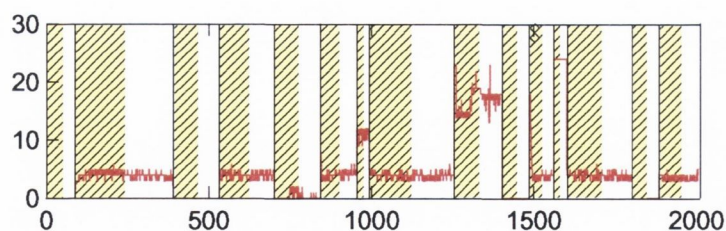


Figure 6.8: The training vector for *Higgins* is the hatched areas of the plot. Frames are on the abscissa and the number of quantised levels are on the y-axis.

in table 6.4 for snooker and table 6.5 for tennis. Bar charts in figure 6.9 show the average performance of the retrieval in terms of precision and recall over the snooker and tennis footage for the individual shape and colour features.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	76.29%	83.12%	79.19%
Precision	86.76%	90.39%	84.66%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	79.32%	84.91%	85.65%
Precision	85.68%	86.12%	93.65%

Table 6.4: Classification results. Mean precision and recall using shape and colour moments for all snooker sequences.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	78.50%	79.33%	77.37%
Precision	89.08%	86.66%	92.34%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	83.5%	82.28%	86.13%
Precision	80.69%	83.65%	88.26%

Table 6.5: Classification results. Mean precision and recall using shape and colour moments for all tennis sequences.

Tables E.1-E.3 provide the precision and recall results for the snooker footage and tables E.4-E.7 of appendix E give the tennis results for the relevant shape and colour moments.

6.4.3 Comments on classification and improvements by merging the results

To enhance the performance of the view classifier, the results from the individual shape and colour classifications were merged by cascading two classifiers. This was achieved by first considering the classifications due to the colour moment feature. This feature proved to give the best classification for the majority of sequences.

For shots which remained unclassified using this method the results from the remaining shape feature classifications were used. The observation sequences which are unclassified arise from the inability of the HMM to resolve the observation sequence with a particular

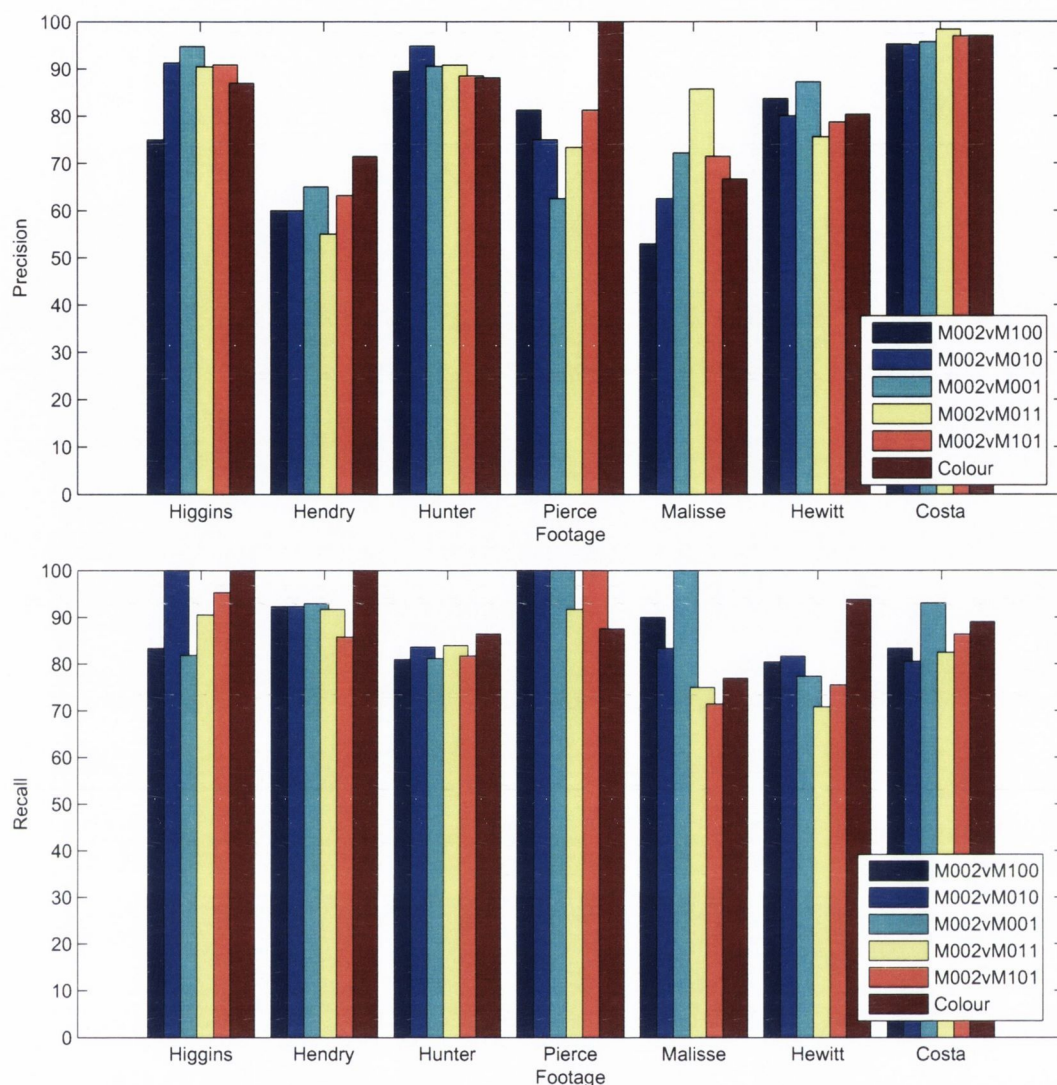


Figure 6.9: Bar charts of the average precision (top) and recall (bottom) for the individual shape and colour features for all snooker and tennis footage.

view model. This is because the likelihood of those particular observation sequences, given each view model is zero. Since the HMMs are only trained on half of each shot, in some cases the remainder of the sequence is subject to some global motion which alters the observations sufficiently so as to not recognise them as being attributed to any model. Results from the colour classification which had a likelihood less than that achieved using the shape features

were also replaced with relevant classifications obtained by that shape feature (equation 6.14).

$$\hat{\lambda} = \arg \max_{1 \leq c \leq C} \left[P(\mathbf{X}^{colour} | \lambda_c^{colour}), P(\mathbf{X}^{shape} | \lambda_c^{shape}) \right] \quad (6.14)$$

The new results of the classification are tabulated in tables E.8-E.10 for snooker and tables E.11-E.14 for tennis in appendix E.

Recognition results achieved by merging the shape and colour feature classification results have shown to give an improvement, for most cases, over the use of the features individually. Misclassification arises due to the similarity between some of the different shot types. For example, another view of the playing area can exhibit similar colour and shape content to that of the global view.

The average performance of the classification for each combination of features is given in table 6.6 for snooker and table 6.7 for tennis. The plots in figure 6.10 show the average retrieval performance for all footage sources using a combination of colour and shape features.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	90.88%	94.20%	86.74%
Precision	88.80%	90.32%	90.56%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	91.88%	92.75%	
Precision	89.30%	90.16%	

Table 6.6: Classification results. Mean precision and recall using a combination of the colour and shape moments for all snooker sequences.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	83.98%	82.99%	87.89%
Precision	92.68%	92.11%	94.45%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	82.26%	84.60%	
Precision	93.54%	92.22%	

Table 6.7: Classification results. Mean precision and recall using a combination of the colour and shape moments for all tennis sequences.

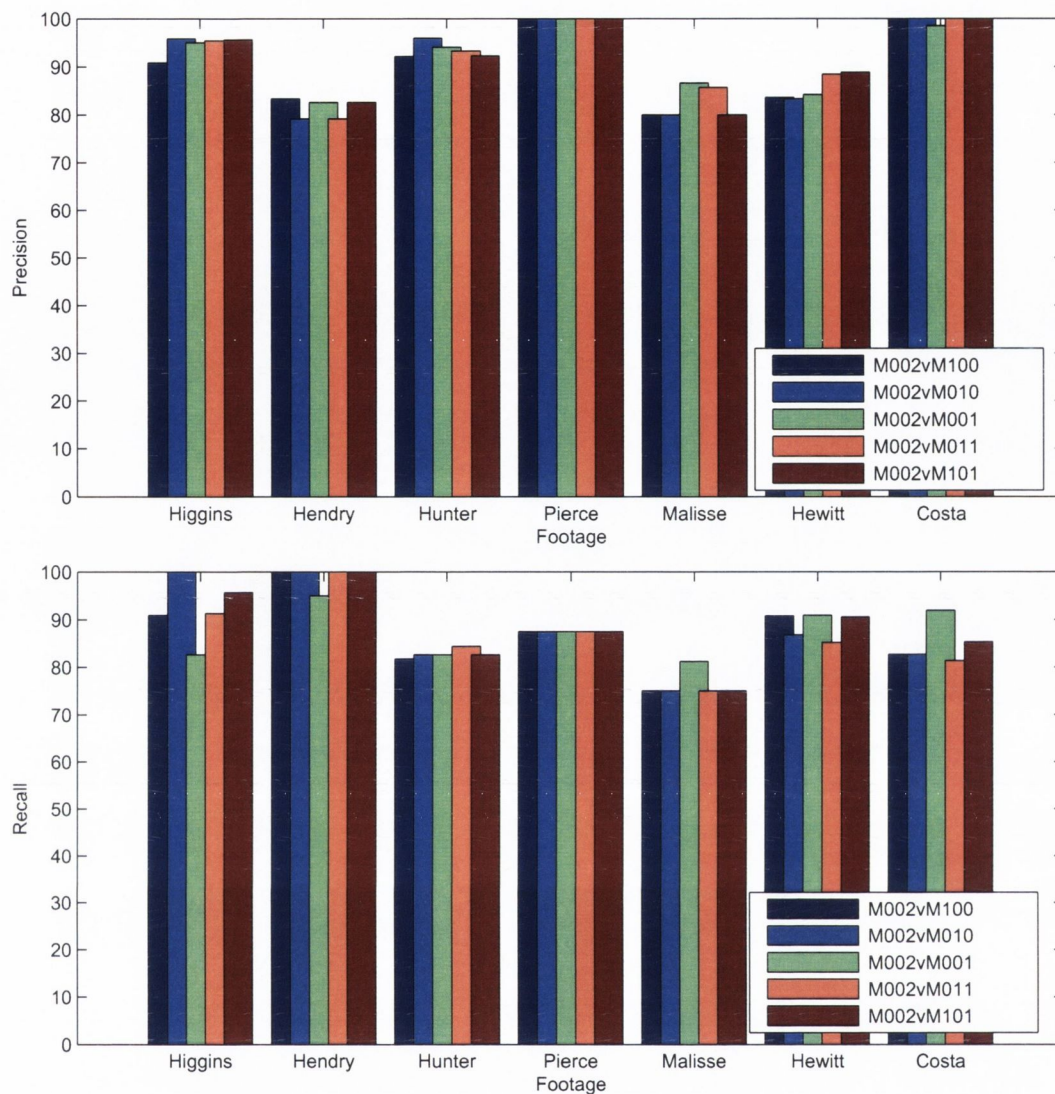


Figure 6.10: Plots of the average precision (top) and recall (bottom) for a combination of colour and the relevant shape features for all snooker and tennis footage.

Alternate training sequence on longer sequences

The training performed previously does not reflect true classification which might be needed in practice, but since only short sequences were available it was necessary to train in this way. The longer sequences were assessed again by training the view models with only half the shots in the sequence. So, using figure 6.8 as an example, only the training up to frame 1000 would be used. The classification results for *Hunter*, *Hewitt* and *Costa* are tabulated in

tables 6.8, 6.9 and 6.10. Bar charts of the retrieval performance are shown in figure 6.10.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	81.74 %	82.61 %	82.61%
Precision	92.16 %	95.96%	94.06%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	84.35 %	82.61%	
Precision	93.27 %	92.23%	

Table 6.8: Results of the classification using a combination of colour and each of the shape features for the Hunter sequence using half the shots for training.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	90.74%	86.79%	90.91%
Precision	83.64%	83.33%	84.21%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	85.19 %	90.57%	
Precision	88.46%	88.89%	

Table 6.9: Results of the classification using a combination of colour and each of the shape features for the Hewitt sequence using half the shots for training.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	82.67 %	82.67%	91.89%
Precision	100%	100%	98.55%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	81.33 %	85.33%	
Precision	100%	100%	

Table 6.10: Results of the classification using a combination of colour and each of the shape features for the Costa sequence using half the shots for training.

As expected, when compared to the averages classifications in section 6.4.3, recall results are generally not quite as good as those achieved using the previous training sequences. This

could be a result of some of the shots which were not used for training exhibiting different camera motion resulting in missed classifications. However, classification remains reasonably good. This supports the argument for using the robust HMM for classifying observations with such stochastic properties.

It is important to consider that this method of training could be compared to training from one game and testing on another within the same tournament. This approach would be useful for a broadcaster who might require view classification over the duration of a tournament (for example the BBC are providing coverage for the 2005 Embassy World Championship which will have 31 games with a minimum of 10 frames in each game).

Training and testing on different sequences

When modelling the evolution of the features with an HMM, it is desirable that the HMM model parameters estimated from one broadcast of a particular sport could be used on another broadcast of the same sport. In the framework used here it is not straightforward to exercise this aspect of model based estimation. The reason is that to use the HMM, the input multi-dimensional continuous feature space is quantised to create a single dimension data stream with samples represented by an alphabet of a fixed number of clusters. These clusters correspond to the different Gaussians used in the GMM model estimation process. Consider two outdoor games of tennis *e.g.* Wimbledon played and the Stella Artois Championships which are both played on grass. Because the features used are based on the colour content of the scene, the colour description of one game may not be appropriate for another game even though *semantically* the objects in the game are the same. Thus grass in Wimbledon for instance does not look the same as grass in New York. This problem is worse for games played in totally different arenas where the colour and materials and even the broadcaster is changed.

These effects imply that the meaning of the symbols used in the alphabet for an HMM appropriate for one game, are not the same for another game. Put another way, there is no reason for the GMM to select the same symbol for the red in a clay court match and the green in a grass match. Therefore, to make the HMM in one game be applicable to the HMM in another a system must be put in place to attempt to make the quantised feature space have the same number of entries for the same type of event *and* make those entries correspond to the same **semantic** feature.

While this thesis does not consider this far ranging implication, the attempt was made to explore these issues with a further view classification experiment. In that experiment, by training on one sequence (*Hendry*) and testing on a longer sequence from a different broadcaster (*Hunter*). Again, as the colour content generally differs from broadcaster to broadcaster, labelled clusters in one source will not correlate with those in the other. It is therefore necessary to map the features between the sequences to obtain a reasonable

observation sequence for classification.

The first mapping that can be easily assigned is the global view. That view occurs most frequently and colour content remains fairly consistent throughout. This is corroborated by table 3.1. The greatest mixture weights from the GMM clustering in the two sequences should therefore correspond to the global view. The likelihood of each data pair within a distance of 0.1 from each mixture component in feature space with weight greater than 0.25 is therefore calculated. These points are then considered to be part of the global view clusters. Close up views of the table exhibit more green content than red, while close ups of the players exhibit more red than green. Separate clusters are assigned to each of these views based on this observation. The remaining two views are significantly more difficult to distinguish in feature space. They occur less frequently and have similar feature values to the more prevalent view types. The clusters corresponding to these views were labelled by observation. In this experiment, the number of clusters is therefore reduced to five for each of the footage sources to maintain a consistent number of clusters for testing.

A classification recall of 100% was achieved and a precision of 68.7%. The low precision is a result of the feature mapping from which some views are incorrectly labelled.

This issue of ensuring the consistency of the quantisation is crucial for generalisation of the HMM between different broadcasts and should be the focus of future work in this area. Note however that this observation does not invalidate the work done here, since even if a different HMM has to be trained for different sequences, the point is that the data used for training is always a tiny fraction of the length of the entire broadcast.

6.5 Event Classification

In [5, 24, 76], the temporal interleaving of camera views was found to have inherent meaning in terms of the various semantic events. By classifying the different camera views using low-level content based features such as colour ratios and global motion, a model could be created for high-level events. In [76], to create the model for a 'service break' for example, it was observed that a non-global view was followed by a dissolve transition. This in turn was followed by a shot of the full court or another non-global view. The model is terminated by a state which represents a close-up of the player. Separate states in the HMM are assigned for each of these views. The states are interconnected by a left-to-right topology representing the temporal evolution of the views. Using a quantised feature vector derived from the low-level content, a separate HMM is trained for each semantic event.

In broadcast snooker footage however, it seems as though the editorial arrangement of camera views is independent of the semantic event which has occurred. This is also the case for some events that occur in tennis which cannot be inferred using the methods outlined in Kijak et al [76], such as a serve and volley. Furthermore, there is a reliance on the editor to comply with editorial conventions. This might not occur from broadcaster to broadcaster

and from year to year.

Other features from the footage must therefore be discovered that *do* convey some semantic information. By considering that the spatio-temporal evolution of objects as they traverse the playing areas embody high-level information, the well known on-line character recognition paradigm can be applied to the sports domain. This approach can be validated by juxtaposing the arbitrary strokes made to draw out characters, with the movement of the objects which bear out high-level events.

On-line handwriting recognition

On-line handwriting recognition typically relies on sampling the trajectory of a stylus as it inputs characters on a pressure sensitive pad. The features extracted from the input are generally a combination of direction, pressure (pen up/down), angle between two consecutive samples and location from a star shaped quadrant quantisation [142]. The evolution of the features are commonly modelled using a left-to-right topology [65, 142] where each state is representative of a character stroke (the quantised stroke between a pen up and pen down).

Since each stroke corresponds to a state in the model, on-line character recognition systems are capable of handling different stroke orders since various topologies can be assigned to the various ways that a human might write a character [85]. This encourages its use for modelling different plays performed by the players in sports since each play, while having the same semantic interpretation may undergo different evolutionary paths.

Sport Applications

This research approaches semantic event recognition using explicit tracks of objects which are deemed to be important and carrying relevant semantic information as outlined in sections 4.4 and 4.5 for snooker and tennis respectively. These are in turn, used as observation features in order to model high-level events which occur in sports footage under a HMM framework.

Application to snooker: It was observed that the position of the white ball at any time instance can allow one to deduce particular events occurring in the footage. The spatio-temporal behaviour of the white ball, over the duration of a player's shot is considered to embody a semantic event. A shot is the time from which the white ball starts its motion until all the balls being tracked come to rest or are potted. Pot detection can be used as a binary classifier to distinguish between certain events when ambiguity is present.

Application to tennis: Tennis footage also offers the prospect of relating the spatio-temporal behaviour of objects to high-level concepts. For broadcast tennis footage, it is very hard to track the ball as it travels at great speeds and deforms as a result of motion blur. This invariably leads to the possibility of tracking the players. As tracking using the

particle filter has shown good results for tracking both players, it offers the prospect of extracting rich semantics directly from the footage. An event in tennis is therefore embodied by the motion of the player in the time period between the detection of the first racquet hit and a transition from a global view to a non-global view. Court view detection and initial player locations are used as binary classifiers if there is a discrepancy between models.

6.5.1 Spatial encoding of the playing area

The playing areas in tennis and snooker can be spatially encoded using the playing area finding algorithm outlined in sections 3.3.2 and 3.3.1 respectively. By consecutive subdivision of the relevant sections, regions can be labelled even in the presence of global motion. The spatial encoding of the playing area also leads to a simple procedure for assigning the initial and state transition probabilities in the HMMs for each high-level event. The following sections detail the spatial segmentation for both sports.

Discretisation of snooker table positions

The dimensions of the table (figure B.1), the positions of the balls and their values dictate the flow of the play to be mostly along the long side (from the baulk area to the black spot)⁴ of the table. The vertical position of the white ball over the duration of a player's shot, could therefore be considered exemplify a particular semantic event.

Using the fact that diagonals of a trapezoid intersect at its centre, the table can be divided into 5 horizontal sections at the coloured balls spot intervals (figure 6.11). Initially, the table is divided by intersecting the main diagonals, retrieving the centre line. Sub division of the two resulting sections retrieves the pink and brown lines, and so on. The starting and end positions of the white ball alone do not sufficiently represent a semantic event. The model must be augmented by the dynamic behaviour of the ball. The observation sequence, \mathbf{X} , is therefore the sequence of evolving table sections.

Discretisation of tennis court positions

In a similar fashion to that used for snooker, the tennis court is divided into sections. The existing delineating field lines are used to provide an initial spatial segmentation of the court. These quadrants alone are not sufficient to represent the position of the player at all times as they can sometimes move out of the court region. These regions must also be accounted for. Figure 6.12 illustrates the spatial segmentation of the tennis court. It was decided to divide the court into 24 segments where each segment is associated with the lines of play on the court. The value of 24 was considered to give a sufficient discretisation of the space to

⁴Without loss of generality, this dimension is assumed to be the vertical, in that the full table view is usually broadcast as such.

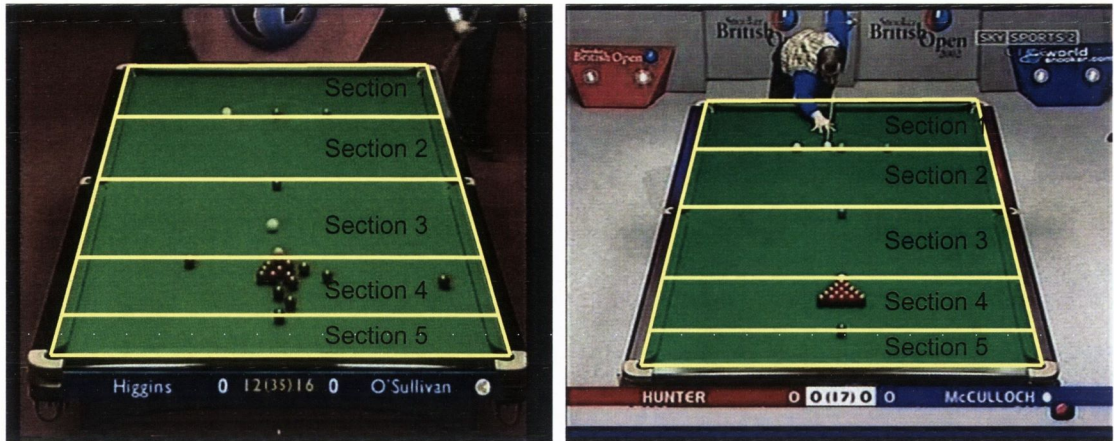


Figure 6.11: Spatial segmentation of the table into 5 sections.

enable the determination of the various events required. The figure on the right hand side of figure 6.12 shows the typical close up view of the the global view. The figure on the left is just used for illustrative purposes to show all 24 quadrant of the court.

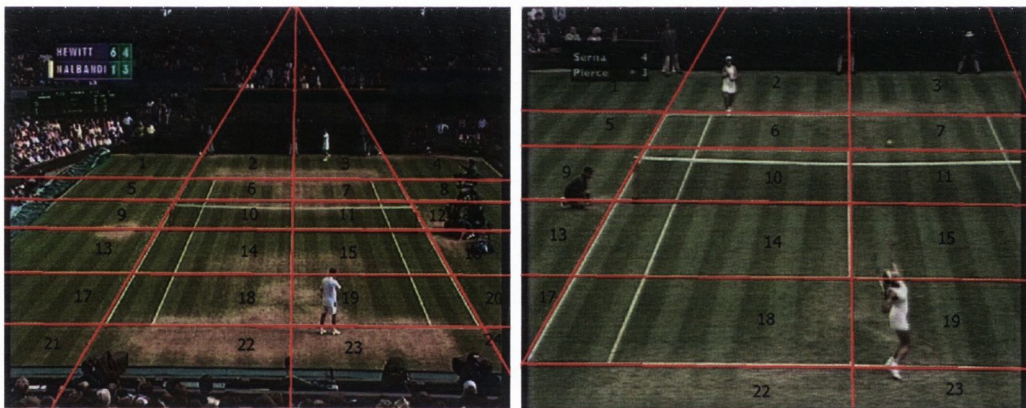


Figure 6.12: Spatial segmentation of the tennis court into 24 sections.

6.5.2 Parsing the footage at an event level

For the purpose of this research six well known events from snooker are considered along with five from tennis.

Snooker

The events considered in this thesis were established by querying 6 amateur snooker players⁵ as to which events they thought to be most important in the game of snooker. Break building, conservative play, snooker escapes, shot to nothings, open tables and fouls were chosen. These events are outlined below and examples of the events are illustrated in figure 6.13.

Break building: When break building (illustrated in figure 6.13 (top left)), the player will attempt to pot the balls in a red-colour-red sequence, thereby accumulating the highest score possible. This is generally accomplished by keeping the white in the middle of the table amongst the red balls. The balls are generally potted in the easier bottom corner pockets when attempting to build a large break. In terms of the section observations this will mean that the white ball will be prominently contained in section 3 or 4. Occasionally a ball will need to be potted in either of the middle pockets depending on the positions of the remaining balls. This could mean that the white ball will have to traverse the table and use the top cushion to regain a good position on the reds.

Conservative Play: During the game each player will attempt to make his opponent's shot as difficult as possible if there is no easy shot available to him. In this case, the player will endeavour to play the white ball in such a way as to place it in the baulk region behind the three coloured balls (yellow, brown and green) or close to the bottom cushion so that his opponent is 'snookered'. This will generally be the state of play at the start of a frame as both players jockey for position and try to force his opponent into making an error. An example keyframe of a conservative play is illustrated in figure 6.13 (top right).

Escaping a snooker: As is required by the rules of the game, the player must attempt to hit a red ball on his first shot after play has been passed to him. If he is snookered he will attempt to either nestle the white amongst the reds, send the white ball back to the baulk region, or make the next shot for his opponent as difficult as possible. A snooker escape is shown in figure 6.13 (middle left).

Shot-to-nothing: When attempting to snooker ones opponent, if the shot has not been played well, a red ball might be in a position to be potted by the opponent. The white ball might be situated close to the top of the table in a similar fashion to the conservative play model, but on this occasion the player might attempt to pot a ball instead of playing safe. This model will be the same as that used for conservative play as the player will attempt to return the white to the baulk area so as not to leave his opponent with an open table. In this case however, the detection of a ball pot is

⁵These are the same players used to establish the training observations outlined in section 6.5.3.

able to disambiguate between both events. An example of a typical shot-to-nothing is illustrated in figure 6.13 (middle right).

Open Table: An open table (figure 6.13 (bottom left)) occurs if a player is attempting to build a break and misses a pot. This is because, as discussed previously, the player will generally try to keep the cue ball in such a position so as to increase his break (*i.e.* in the middle of the table). If he misses, his opponent will have the opportunity to take his shot from a promising position and hence is left with an open table.

Foul: In the case of each event, if a new track is not instantiated by the collision of the white ball with a colour, a foul is flagged. A second condition for a foul to be declared is if the white ball is potted. A ball potted in the incorrect sequence will also result in a foul being called. However, even though the tracker allows the colour of the ball which has been potted to be detected, pots might be displayed in camera views other than the global view. Consequently, the sequence of pots cannot be accurately recorded. Such fouls cannot be detected in this event detection implementation. A foul is shown in the bottom right of figure 6.13.

Tennis

Similar to the snooker footage, 6 tennis players were asked their opinions on the most important events which occur in tennis. Aces, faults, double faults, attacking plays (serve and volley) and rallies were chosen. It is difficult to illustrate the events using a keyframe such as those used for snooker but the descriptions below should suffice.

Ace: From their first service, a player will generally attempt to hit the ball so that their opponent will be unable to return the ball, or hit the ball with any part of their racket. In this case, the player serving will stay in the same vicinity as which they started their serve, not moving across the court or too close to the net. The camera view then cuts to some other view, normally a close up of the player. To supplement detection of the event, audio information [31] can be used. There should only be one audible hit before the camera cuts to different view.

Fault: If from the first serve of a player, the ball is hit either into the net, or outside the bounds of legal play for that particular serve, a fault is incurred by the serving player. The player then moves back to the service line for their next serve.

Double fault: If a player incurs two faults on the same service, a point is awarded to the other player and a double fault is called. This event can be detected using combinations of the ace and fault events and knowledge that the player should be serving from the same side of the court. It can also be detected by its own particular motion attributes, if the event is conveyed in one contiguous shot.



Figure 6.13: Illustration of the high level events in snooker. Top: Break building (left), Conservative play (right); Middle: Snooker escape (left), Shot to nothing (right); Bottom: Open table (left), Foul (right).

Attacking serve and volley: Following a good service, the serving player might attempt to win the point by moving into the net and returning his opponents shot. A shot where the player moves in such a manner is considered to be attacking play.

Rally: After a serve, the players move around the court hitting the ball to one another before one attempts a winner.

6.5.3 Model training based on human understanding of the events

Training data was collated using human understanding of the events. This was achieved using a GUI which allows the user to trace the route which they believe the object should travel for each of the events. Figure 6.14 shows screen shots of the GUI for tennis and snooker with training tracks in each for the event listed. There are three principal benefits of training models based on human perception gathered in this way.

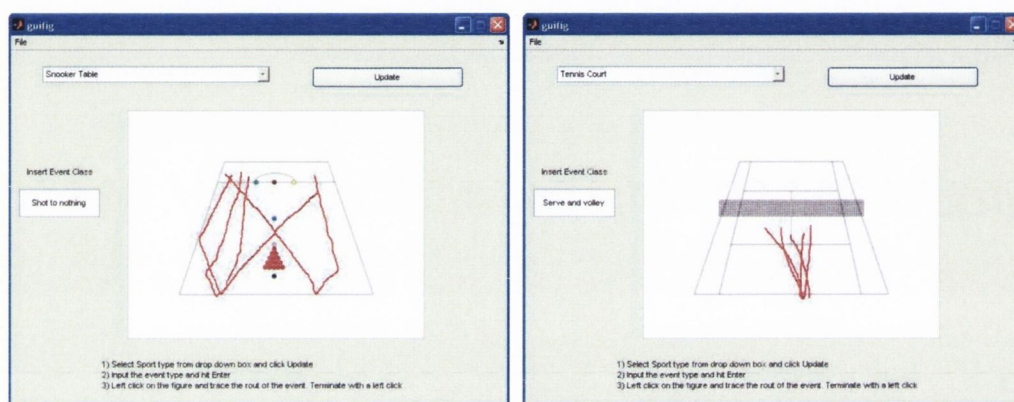


Figure 6.14: GUI used for user training. Left: Training a shot to nothing event in snooker with three tracks. Right: Training a serve and volley event from the right in tennis with five tracks.

- Unlike abstract observations such as human feature based representations [119] or motion energies [161], the observations used have an inherent meaning to the user, and are well understood in terms of the geometrical layout of the playing area.
- Since human training is used, there is no need to parse vast amounts of training data in the corpus.
- From the user perspective, if he/she is not happy with the training provided, it would allow them to tailor the retrieval for their own viewing purposes.

The training is provided with the knowledge that high-level events in snooker and tennis are punctuated by certain constraining observations. In snooker, this constraint is that the event occurs in the time period between which the white ball begins its motion until all balls being tracked come to rest or are potted. In tennis the motion of the player in the time

period between the detection of the first racquet hit and a transition from a global view to a non-global view embodies the semantic event.

Snooker Observations

In order to accurately correlate the notion of the events listed to the temporally evolving position of the white ball, six amateur snooker players were independently asked to express their perceptions of each event in terms of the spatio-temporal position of the white ball using the GUI. They were asked to make 5 tracks for each event. This allows the formulation of models which reflect several human opinions.

The question was posed as follows:

“In terms of the motion of the white ball, in what sequence of sections ⁶ would it need to traverse in order to represent each of the following high-level episodes: Attempting a snooker, escaping a snooker, shot-to-nothing, break building. Furthermore, what events (ball pots or misses, for example), would also have been experienced by the white ball or other balls on the table.”

A single realisation of the each of the players opinions are tabulated in table 6.11. Similar answers were given for most of the events by each player (see figure 6.12 for the spatial segmentation of the snooker table).

	B.B.	Conservative Play	Snooker escape	Shot to nothing
Player 1	[3, 4, 3]	[1 → 5 → 1]	[1 → 4]	[1 → 5 → 1]
Player 2	[3, 4, 5]	[1]	[1 → 5]	[1 → 5 → 1]
Player 3	[3]	[2 → 5 → 1]	[1 → 5]	[1 → 5 → 1]
Player 4	[4, 5, 4]	[2, 1]	[1 → 4]	[1 → 5 → 1]
Player 5	[4, 5, 4]	[5]	[5 → 1]	[1 → 5 → 1]
Player 6	[4, 3]	[4, 5]	[1 → 5]	[1 → 5 → 1]

Table 6.11: Player’s judgement on the sequence of sections needed to exemplify an event. The notation $a \rightarrow b$ implies that the ball traverses all regions from section a ending in b . Illustrations of a break building event and a shot-to-nothing are shown in figure 6.13.

As can be seen from the table, the requirements for conservative play are similar to those of the shot-to-nothing. However, the pot classifier can be used to distinguish between the two events as the shot-to-nothing requires that a pot occurs while conservative play does not.

⁶The concept of the spatial encoding of the table had already been explained

Tennis Observations

A similar question was posed to six tennis players. They were asked how the motion of the serving player should be reflected in the event of an ace, fault, double fault, rally and attempted attack or serve and volley using the GUI. Again, they were asked to make five tracks. Note that the observations considered here only take into consideration the behaviour of the player in the lower half of the court but can be easily adapted for a player in the top half (see figure 6.12 for the spatial segmentation of the court). Furthermore, using this method of high-level event classification, the events can be further discriminated into having originated from the left or right (for faults, ace and double fault) hand side of the court. This could prove useful for coaching videos where a serve from one side of the court might be weaker than the other. Table 6.12 shows a realisation of each event from the six players where each event originates from the right hand side of the court.

	Ace	Fault	Double Fault
Player 1	[23, 19]	23	[23, 19, 23]
Player 2	23	23	[23, 22]
Player 3	[23, 19, 18]	23	[23, 19, 18, 22]
Player 4	[23, 19]	[23, 19, 18, 19, 23]	[23, 18, 22]
Player 5	23	[23, 19, 23]	[23, 22]
Player 6	[23, 19]	[23, 19, 18]	[23, 19, 23]
	Rally	Serve and Volley	
Player 1	[23, 19, 18, 22, 21, 22]	[23, 19, 15]	
Player 2	[23, 18, 19, 20, 24]	[23, 19, 18, 14]	
Player 3	[23, 19, 15, 14, 18, 22]	[23, 19, 14]	
Player 4	[23, 18, 22, 21, 22, 23]	[23, 19, 15, 14]	
Player 5	[23, 19, 18, 17, 18, 19, 22]	[23, 19, 15]	
Player 6	[23, 19, 23, 19, 23, 20, 23]	[23, 19, 15, 14]	

Table 6.12: *Examples of the realisations of the players in terms of the evolution of the position of the player in the bottom half of the court. These observations typify a service which originates on the right hand side of the base line.*

6.5.4 Establishing the model topologies

The model topologies are derived from the observations generated by the training perceptions of the users. The topologies therefore reflect the nature of the target patterns for the required events in both snooker and tennis.

Snooker

Events in snooker can be modelled by analysing the spatio-temporal behaviour of the white ball and observations from another ball being tracked. The observations used are:

$$\mathbf{X} = \{\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3, \mathcal{X}_4, \mathcal{X}_5\} \quad (6.15)$$

- \mathcal{X}_1 is the spatially encoded trajectory white ball.
- \mathcal{X}_2 is a binary value which indicates whether the white ball has been potted or not collided with a coloured ball.
- \mathcal{X}_3 is a binary value which is 1 if a global view occurs during a players shot.
- \mathcal{X}_4 takes a binary value which specifies whether the coloured ball that has been hit has been potted.
- \mathcal{X}_5 indicates if the white ball is moving before a global to non-global view transition is detected.

The observations therefore take the form:

$$\begin{aligned} \mathcal{X}_1 &\in \{1, \dots, K\} \\ \mathcal{X}_2 &\in \{0, 1\} \\ \mathcal{X}_3 &\in \{0, 1\} \\ \mathcal{X}_4 &\in \{0, 1\} \\ \mathcal{X}_5 &\in \{0, 1\} \end{aligned} \quad (6.16)$$

Knowing the number of states, where a state is representative of a table section, $N = 5$, and discrete codebook entries, $K = 5$, a model λ_c , can be defined for each of the competing events, \mathcal{C} . The observations that result from the current state of the white ball and the coloured ball are taken as binary classifiers to help distinguish between events with similar models.

The occurrence of a ball pot or miss (the white ball not colliding with a coloured ball) will affect the viewer semantics. Priori domain knowledge allows a set of heuristics to be established which are used to evaluate the current maximum likelihood classification upon detection of a miss or a pot. It was also observed that a snooker escape event is characterised by a cut from the full-table view to a close up view of the ball about to be hit. This occurs while the white ball is still in motion. If the velocity of the white ball is greater than zero, a snooker escape is inferred. The models for each event are specified below.

- Break building (λ_1): The topology (figure 6.15) of this 5 state HMM enforces the occupation of states 3,4 and 5.

If a pot has been detected, $\mathcal{X}_4 = 1$, the player is attempting to build a high break. In the unlikely event of one of the balls not being potted, $\mathcal{X}_4 = 0$, the white ball will

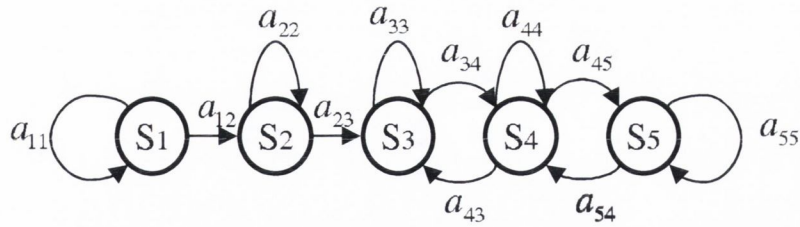


Figure 6.15: HMM for break building.

probably be in a position such that the remaining balls will be eminently ‘pottable’. This is called an **open table** event (λ_5).

- Conservative Play (λ_2): A data driven approach to the design of this topology (figure 6.16) ensures that the states 1 and 2 or 4 and 5 are occupied. If this model is

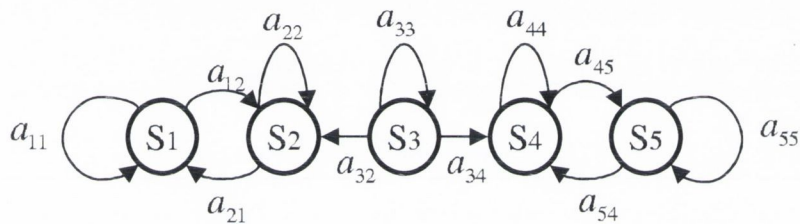


Figure 6.16: HMM for conservative play.

chosen as being the most likely, and a pot is detected $\mathcal{X}_4 = 1$, a shot-to-nothing will be inferred (λ_4). This is because the ball will be in an area where it might prove difficult for a player to pot the next coloured ball in the sequence. If there is no pot $\mathcal{X}_4 = 0$, the model choice remains the same (λ_2). This is a sufficient model for conservative play, even though some of the training provided by the players is similar to that used for the shot-to-nothing model. The lack of a pot, even if that model is selected will infer a conservative play (see the flowchart in figure 6.18).

- Escaping an attempted snooker (λ_3): Since a player can be snookered from either end of the table a left-to-right, right-to-left topology (figure 6.17) is required to encompass all possible eventualities of the event as given by the observations in table 6.11. If a pot is detected following the classification of a snooker escape $\mathcal{X}_4 = 1$, the heuristics will infer a break-building event (λ_1). As the only goal of the player will be to escape the snooker without conceding a foul or an open table if a ball is potted, it simply serves as a bonus. If there is no pot, $\mathcal{X}_4 = 0$, the classification will remain the same (*i.e.* a snooker escape (λ_3)). It has also been noticed from the footage that a snooker escape

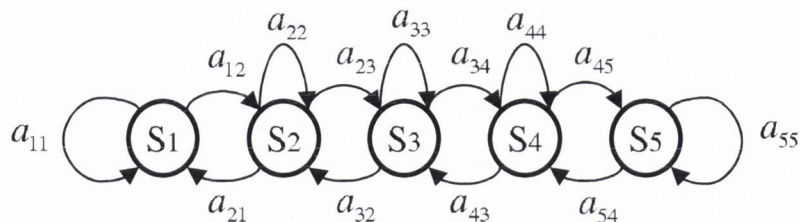


Figure 6.17: HMM for a snooker escape and a shot-to-nothing.

is also stylised by a view transition $\mathcal{X}_3 = 1$ while the white ball is moving $\mathcal{X}_5 = 1$.

- Shot-to-nothing (λ_4): From observing the training data in table 6.11, a shot to nothing can be modelled using a 5 state, left-to-right, right-to-left HMM (figure 6.17). If a pot is detected, $\mathcal{X}_4 = 1$, the pot heuristics will infer a shot-to-nothing (λ_4). If there is no pot, $\mathcal{X}_4 = 0$, the spatio-temporal evolution of the white ball position will show that the player is attempting to return the white ball to the top of the table. A conservative play event, (λ_2), could therefore be inferred as he is making the next shot as difficult as possible for his opponent.

In all of these cases a miss by the white, $\mathcal{X}_2 = 0$, flagged by a non-instantiated second track, or if the white is potted will result in a **foul** (λ_6) being inferred. Play will then be transferred to the opposing player. Furthermore, if a ball-cushion bounce is detected before an inter-ball collision, the player is attempting a difficult shot which will generally be a snooker escape. Both of these classifiers override any ML model selection. Figure 6.18 illustrates the process of determining the correct model for each event.

Tennis

Events in tennis can be modelled by analysing the spatio-temporal behaviour of the players combined with audio information (the racquet hits are detected using the method outlined in Dahyot et al [31]) and initial player locations. Since broadcast footage is being used, the models must be adapted to incorporate view transition detection as a further binary classifier.

$$\mathbf{X} = \{\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3, \mathcal{X}_4\} \quad (6.17)$$

- \mathcal{X}_1 is the spatially encoded trajectory of the serving player.
- \mathcal{X}_2 is a binary value which is equal to 1 when the first racquet hit which occurs in each global view is detected and remains at 1 until a view transition is detected.
- \mathcal{X}_3 is a binary value representing which side of the court the player is on for their detected service (left 0, right 1).

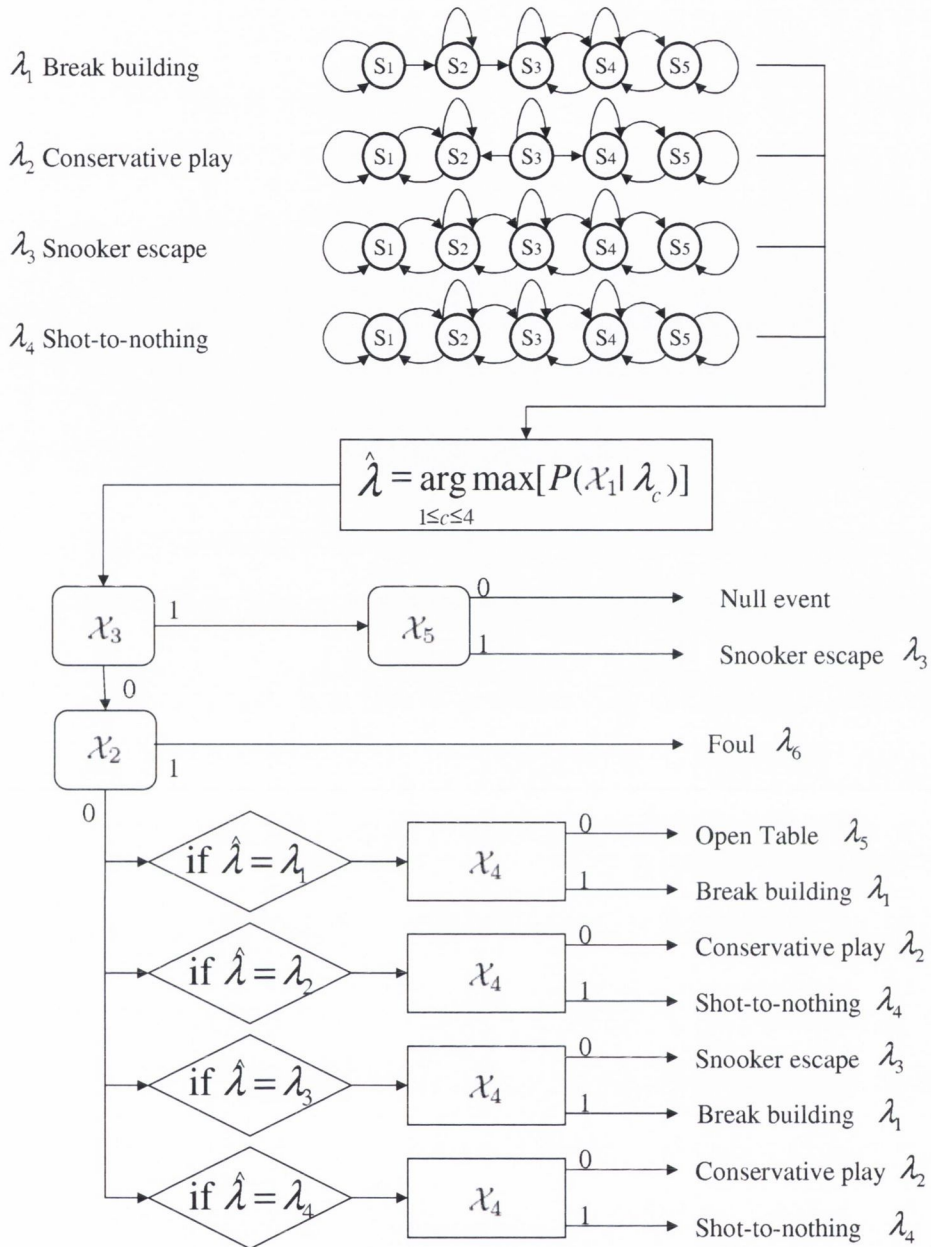


Figure 6.18: Event HMMs with binary classifiers. An event is taken as the time between which the white starts moving and all balls being tracked come to rest. The null event might occur if a player walks in front of the white ball and then uncovers it. A tracker will be instantiated. However, if the white ball does not move until a view transition is detected, it is deemed to be a null event.

- \mathcal{X}_4 is a binary value which is 1 if the time between the first and second racquet hits in a global view is greater than 5 seconds.

The observations take the form:

$$\begin{aligned}\mathcal{X}_1 &\in \{1, \dots, K\} \\ \mathcal{X}_2 &\in \{0, 1\} \\ \mathcal{X}_3 &\in \{0, 1\} \\ \mathcal{X}_4 &\in \{0, 1\}\end{aligned}\tag{6.18}$$

Knowing the number of states, where a state represents a vertical column of the tennis court (see figure 6.12 for illustrations), $N = 4$, and discrete codebook entries, $K = 12$, where each of the observations is a quadrant of the tennis court in the upper or lower half of the court depending on the side from which a service is taken ⁷, a model λ_c , can be defined for each of the competing events, \mathcal{C} . $K = 12$

The models for each event are specified below and are given for a service which originates from the bottom half of the court. The figures (6.19-6.21) attempt to illustrate the possible outputs which might be observed from each state for the particular event type. This is expressed via the rectangles which show the probabilities of the observations from each state. The observations obey the standard constraint that the sum of the observation probabilities for each state is equal to one. The methodology for the choice of event given the observations is clarified in the flowchart of figure 6.22.

1. If $\mathcal{X}_2 = 1$:

- Ace (λ_1), Fault (λ_2): If a single racquet hit is detected $\mathcal{X}_2 = 1$ in the global view, a fault or an ace is deemed to have occurred. The model used for the two events is the same, but the observations differ. For an ace, the player serves and moves to the left or right while for a fault the player moves back to the base line from where they originally served. If two fault events are detected in sequence, with the player serving from the same side of the court for both, the second is considered to be a 'let' (when the ball hits the top of the net but bounces in the correct service box). If a fault is followed by an ace, with the player serving from the same side of the court for both, $\mathcal{X}_3 = 0$ or 1 for both, a double fault is assumed to have occurred. We assume this because the likelihood of an ace on a second service is quite low. A 4 state HMM, which is ergodic for states 2 and 3 is used to model the service and follow up of the player as they move toward the net for an ace, or back to the service line for a fault. The transitions to the left from state 2 and to the right from state 3 model any motion of the player if they walk off the court. The observations probabilities are shown in the rectangles emanating from each state.

⁷The total number of quadrants is 24 but only the behaviour of serving player is considered to embody the event. This means that only 12 quadrants (1...12 for the player in the top half and 13...24 for the player in the bottom half) are needed as observations. Figure 6.12 illustrates the spatial locations of the quadrants.

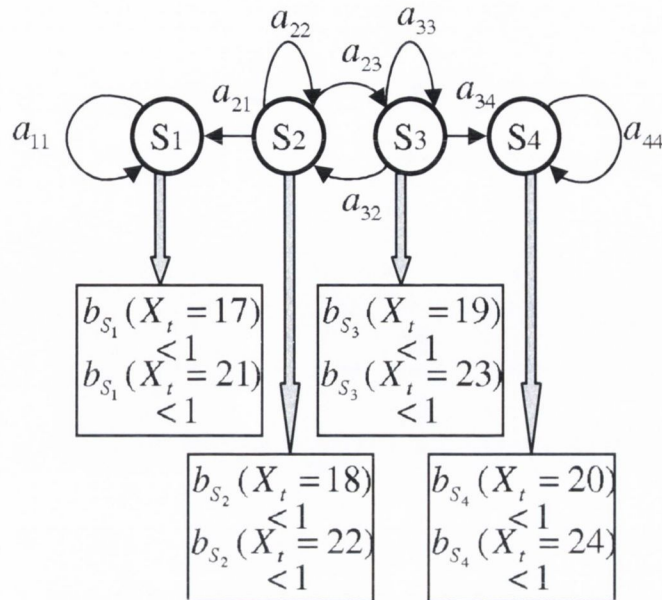


Figure 6.19: HMM for an ace, fault and double fault with relevant observations.

2. If $\mathcal{X}_2 = 0$:

- Fault (λ_2): If $\mathcal{X}_4 = 1$, and on the next detected racquet hit, the initial location of the player is the same *i.e.* \mathcal{X}_3 is the same, a fault is assumed to have occurred.
- Double fault (λ_3): A double fault can be modelled using the same 4 state topology used for the ace and fault events. The different model parameters arise from the difference in the observations (table 6.12). The player makes their first fault, moves back to the base line and then faults again.
- Attacking serve and volley (λ_4): The same topology as is used for the ace and (double) fault is employed for an attacking serve and volley. As the player moves toward the net he generally stays between the two inner tram lines and between the net and the service line⁸. Training observations differ however from the previous events.
- Rally (λ_5): A rally is modelled using a 4 state left-to-right, right-to-left HMM. A rally will generally last several racquet hits and is typified by large amounts of motion around the court illustrated by the realisation of such an event in table 6.12.

⁸See figure B.2 for a labelled schematic of a tennis court.

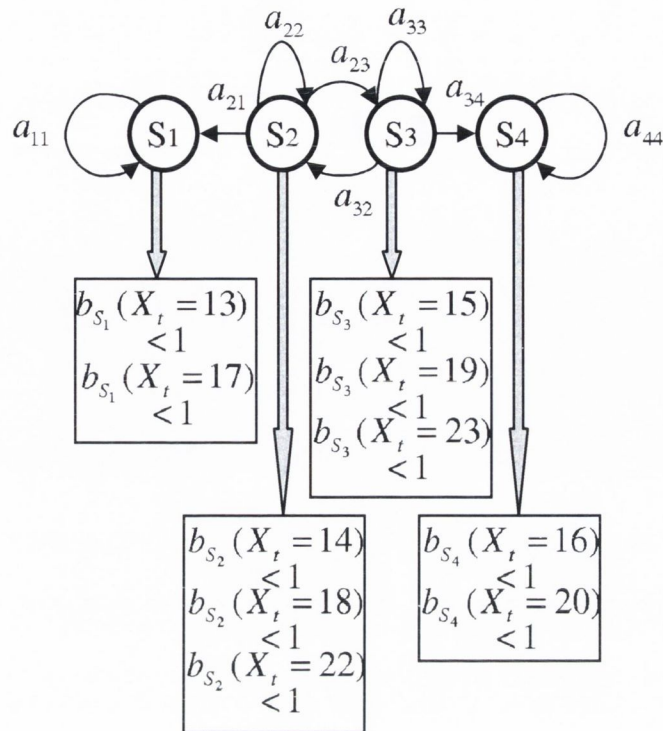


Figure 6.20: HMM for an attacking serve and volley with relevant observations.

6.5.5 Event classification results for snooker and tennis

The goal of this section is to report on the results of high-level event classification in snooker and tennis. The frequency of these events in each of the snooker and tennis footage sources are listed in tables 6.13 and 6.15 respectively.

It should be noted that the sum of the frequency of events for each footage source is not equal to the total number of global views (see table 6.1 for details). This is because more than one event can occur in any global view (*e.g.* a fault followed by a rally, or several break building events in one contiguous shot of the global view). Furthermore, in the same way as other views convey no useful high-level information, if the required objects are not moving while the global view is being displayed, no important information is transmitted. Global views containing no motion information are discarded.

Experiments were conducted on all three sources of snooker footage (*Higgins*, *Hendry* and *Hunter*) and an additional source *King*, from the same broadcaster as *Hendry*. This additional footage was 86622 frames in duration with 109 global views, but was only made available late in the research and was not utilised in previous chapters. Half of this footage was processed using the tracker and the remainder was simulated using the GUI outlined in section 6.5.3. 7 events were observed in *Higgins*, 10 in *Hendry*, 33 in *Hunter* and 116 in *King*.

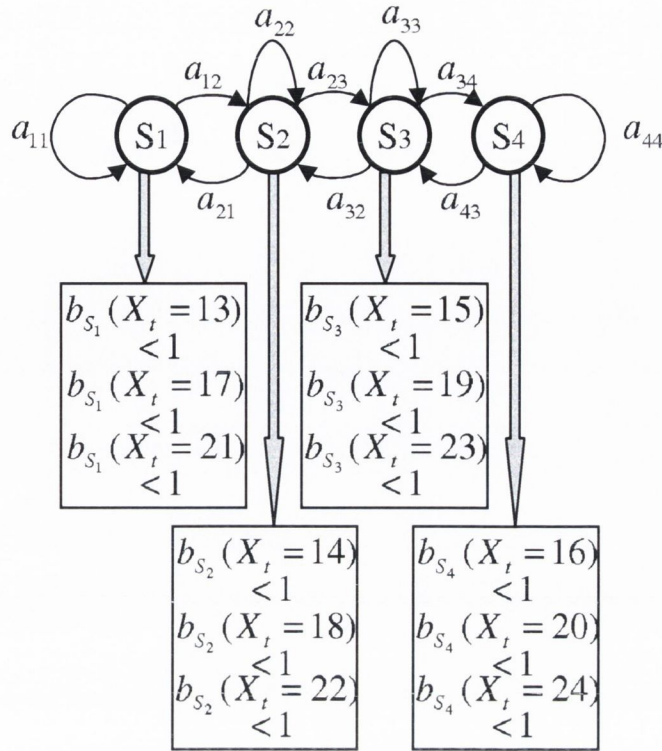


Figure 6.21: HMM for a rally with relevant observations.

Table 6.13 lists the frequencies of these events.

Experiments were conducted on three sources of tennis footage (*Pierce*, *Malisse* and *Hewitt*). Since the geometry was not available in Costa, the players could not be tracked using the method outlined in chapter 4, so the footage was not considered for high-level event detection. 5 events were observed in *Pierce*, 9 in *Malisse* and 20 in *Hewitt*. Table 6.15 lists the frequency of these events.

Given the observation sequence, \mathbf{X} , and the trained models, λ_c , the semantic episode within each clip can be classified by finding the model which results in the greatest likelihood of occurring according to equation 6.19. The binary classifiers are then used to correctly determine the event type. The event classifications results for snooker and tennis are illustrated using confusion matrices in tables 6.14 and 6.16 respectively.

$$\hat{\lambda} = \arg \max_{1 \leq c \leq C} [P(\mathcal{X}_1 | \lambda_c)], \quad \begin{array}{l} C = 4 \text{ events for snooker} \\ C = 5 \text{ events for tennis} \end{array} \quad (6.19)$$

One discrepancy in results is the classification of break building events as shot-to-nothings. The reason for this is in the topology of the break building model. As discussed in the event descriptions, break building is mainly confined to the bottom part of the table. However, it can also occur in the top sections. This is where the inconsistency arises. Some of the shots

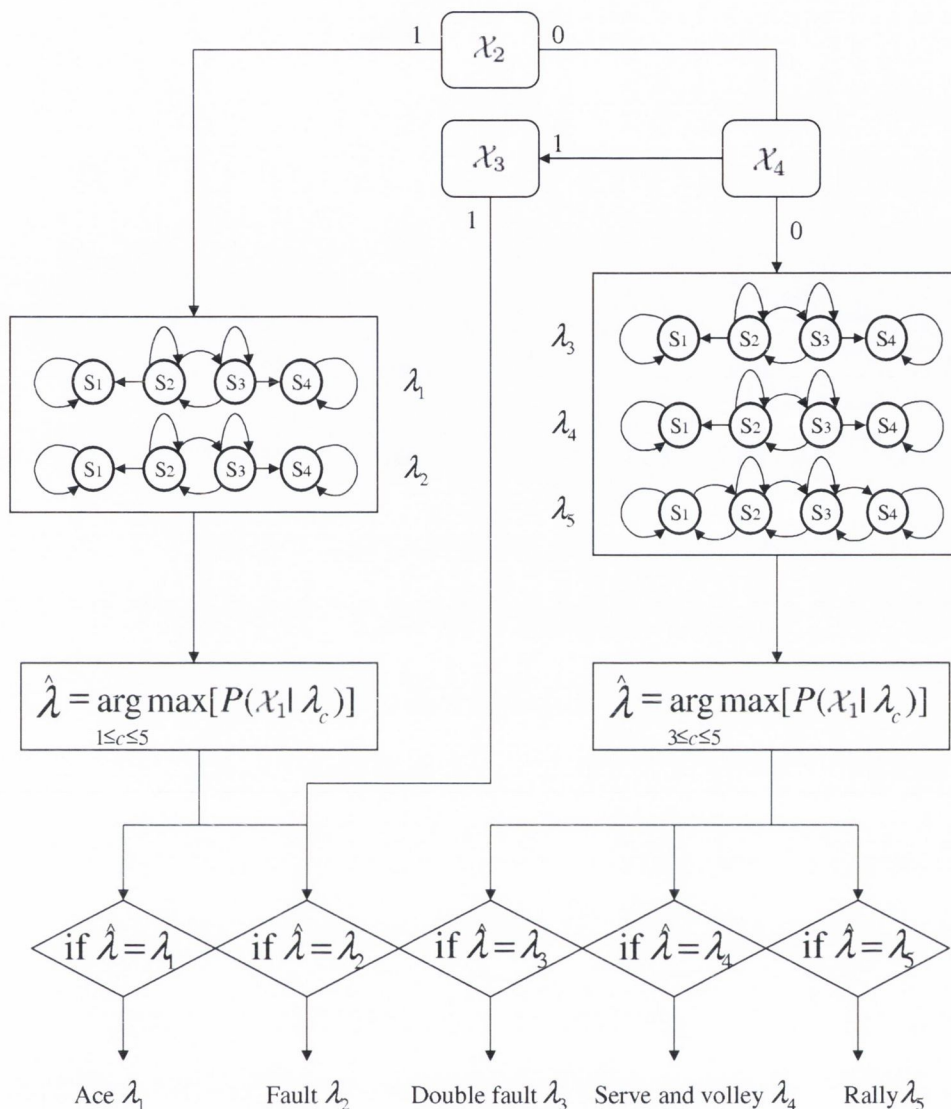


Figure 6.22: Event HMMs with binary classifiers.

taken by the players originate from one end of the table and rebound off the opposing cushion as the player attempts to get a good position on another ball. An illustration of this is shown in figure 6.23. A break building was also classified as an open table on one occasion. This was because a green ball pot could be detected using the ball tracker (figure 6.23). All the break building event which are misclassified as open tables are due to problem in tracking. As discussed in chapter 5, the luminance segmentation does not enable differentiation between ball colours. These separate ball regions appear merged in the binary image. The colour model cannot resolve the target model with the ball to be tracked and pots are not detected.

The missed events arise for a number of reasons. On two occasions in *Higgins* and once

	<i>Higgins</i>	<i>Hendry</i>	<i>Hunter</i>	<i>King</i>
Break building (λ_1)	3	9	22	59
Conservative Play (λ_2)	2	1	6	41
Snooker Escape (λ_3)	0	0	1	3
Shot-to-nothing (λ_4)	1	0	2	5
Open Table (λ_5)	1	0	1	4
Foul (λ_6)	0	0	1	4

Table 6.13: Frequency of events in the global view in snooker.

Event	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	Missed	Total
λ_1	71	1	0	5	5	3	8	93
λ_2	0	37	6	0	5	0	2	50
λ_3	0	0	4	0	0	0	0	4
λ_4	2	1	0	5	0	0	0	8
λ_5	0	2	0	0	4	0	0	6
λ_6	0	0	1	0	0	4	0	5
Total	73	41	11	10	14	7	10	166

Table 6.14: Confusion matrix for event classification in all snooker footage

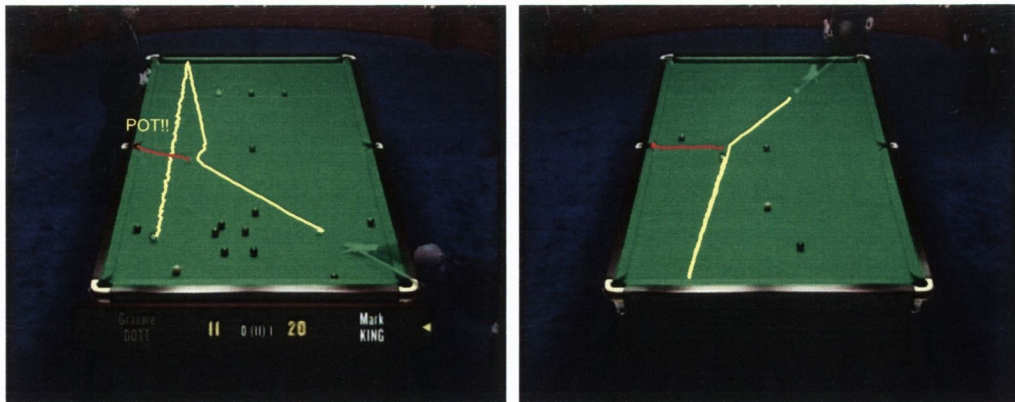


Figure 6.23: Break building classified as a shot-to-nothing (left) and an open table (right)

in *King*, a player attempts to pot a ball along the cushion. Here, the white ball is located very close to the cushion for the duration of the shot and is masked by the player masking. This

means that a track is not generated. Other missed break building events and conservative plays result from a loss of track of the white ball. On these occasions, the player hits the ball with such force that it is out of the range of the tracker. Possible use of a motion proposal in the particle filtering stage would help re-establish the track.

Snooker escapes are confused with conservative plays on occasion due to the track borne out by the ball. A foul is incorrectly classified as a snooker escape in *King*. It arose by the systems inability to determine the correct sequence in which balls should be hit. On this occasion a black was hit instead of a green.

	Pierce	Malisse	Hewitt
Ace (λ_1)	0	1	2
Fault (λ_2)	1	3	4
Double Fault (λ_3)	0	1	1
Serve and volley (λ_4)	0	0	2
Rally (λ_5)	4	4	11

Table 6.15: Frequency of events in the global view in tennis.

Event	λ_1	λ_2	λ_3	λ_4	λ_5	Missed	Total
λ_1	2	0	0	1	0	0	3
λ_2	0	7	1	0	0	0	8
λ_3	0	0	2	0	0	0	2
λ_4	0	0	0	2	0	0	2
λ_5	0	1	1	0	17	0	19
Total	2	8	4	3	17	0	34

Table 6.16: Confusion matrix for event classification in all tennis footage

The tabulated confusion matrix (table 6.16) shows that two rally events are misclassified. In *Hewitt*, one of the rallies is quite short in duration. The player moves from one side of the court to the other where the point is won. This is analogous to the motion behaviour of the player for a double fault event. A similar problem occurs in *Pierce* where a short rally exhibits a comparable track to a fault where the player remains in one state. The final misclassification results from a fault being classified as a rally. In *Malisse*, the single observed ace event is a game point winner. The camera follows the player for a short duration as he returns to his seat. His motion is similar to that of an attacking serve and volley play, and is classified as such.

In both the case of snooker and tennis, under trained models could also be responsible for some of the misclassifications. Additional training should provide better results.

Results using our method (*Rea*) are tabulated in table 6.17 in terms of precision (P) and recall (R), and are compared to those obtained in Kijak et al [75] for equivalent events. The results obtained using our method compare favourably to those obtained by Kijak. It is important to note that our method retrieves an additional three explicit (*i.e.* semantic) events over Kijak while they also retrieve aspects from the footage which do not contain specific semantics (Break in play and Replays) relating to the play.

	<i>Rea</i>		<i>Kijak</i>	
	P	R	P	R
Fault	87.5%	100%	86%	88%
Rally	89.47%	100%	94%	86%

Table 6.17: Comparison between our technique and that obtained by Kijak et al [75].

6.6 Summarisation and Indexing

A browser for viewing high level events which occur in a snooker or tennis match has been created by Andrew Crawford of the Sigmedia group in the Electrical and Electronic Engineering Department in Trinity College Dublin. The browser takes as an input a time indexed file labelled with each event type, the start and end time of the events, the camera view type and the locations of shot cuts for snooker and tennis footage. Supplemental ball colour and pot pocket information are provided for snooker to enable the user to search for pots made by the players by relevant ball colours, pocket numbers or a combination of both. Player position metadata allows the user to discriminate from which side of the court the event originates. A keyframe summary for each high-level event clip is generated off-line to which this data is associated.

A time indexed tree structure of events types, combined with the keyframe summary allows the user to determine if they wish to view the clip. The idea of keyframe selection in this work is to build a synthetic representation of the event using the motion of the objects which convey the semantics in some comprehensible manner. A keyframe for each clip in the snooker footage is generated by superimposing the track borne out by the white ball and the ball which has just been hit on the averaged first and last frame of the clip. An illustration of the keyframes used in the browser is shown in figure 6.24. They are annotated with the event type, and if a pot has occurred, the pocket in which the ball was potted and its colour. High-level events in tennis are summarised using the work of Yeterian [165], also at Sigmedia. A mosaic of the tennis court is rendered, on to which an onion skin of the motion of the player

between detected ball hits [31] is overlaid.

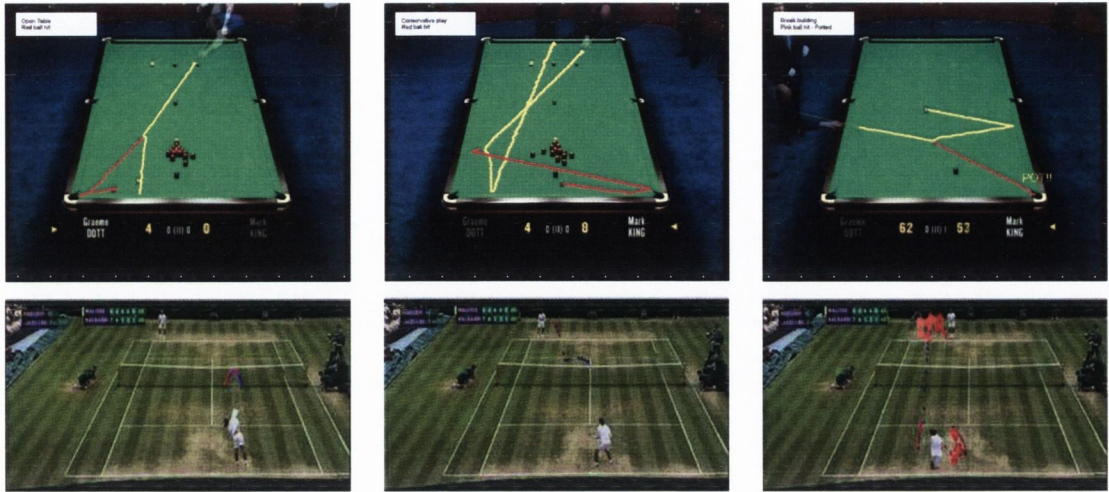


Figure 6.24: Snooker and tennis summaries. Top: A summary of an open table (left), a conservative play (middle) and break building (right); Bottom: Three keyframes from a rally event.

These kinds of keyframe summaries could also prove useful for media aware content adaptation applications where there is insufficient bandwidth to transmit a full video summary of the game. More details on the browser can be found in [29].

6.7 Summary

This chapter presented an approach for camera view and high-level event classification. The HMM was introduced and its abilities to model temporally evolving sequences were presented. A DHMM was chosen to model the stochastic nature of the quantised colour and shape features.

By considering the spatio-temporal behaviour of an object as being the embodiment of the semantics of an event, high-level events were retrieved from broadcast tennis and snooker footage. The treatment of this problem differs from previous works which rely on the temporal interspersion of various camera views to classify the appropriate views. These constraints assume that the editorial arrangement of camera views will be abided by. While there is some requirement for view transitions from global to non-global for detection of certain events in this thesis (because broadcast footage is being used), the techniques described in this chapter are more independent of the editorial process. If only the global view feed was broadcast, the modelling procedures could easily be altered by incorporating other features such as detection of crowd response to capture all the required events, without the need for

these editing constraints.

7

Conclusions and Future Directions

This thesis has presented the means for detecting high-level events which occur in sports using only broadcast footage. A five stage framework for sports video analysis was proposed, with each stage contributing to the goal of detecting semantic events. The difficulty of detecting such events in a video stream has been discussed and the solution provided by this thesis has been shown to be effective for tennis and snooker footage. This was achieved by considering the spatio-temporal evolution of an object as being the embodiment of a semantic process.

In the preliminary stages of the research it became obvious that in order to detect high-level events, it was necessary to firstly simplify this problem to one of detecting where in the video stream these events could actually occur. This initiated the creation of the first two steps of the framework. It was assumed that the global view, which provides an overview of the entire playing surface, was able to capture all the significant events which may be required to create a useful summary of the game. This was affirmed by analysing videos of tennis and snooker footage. The approach taken for detecting this view type was presented in chapter 3. It was argued that sports such as tennis, snooker, badminton, and cricket all occur within predefined playing limits. This means that they are well defined in terms of the geometrical properties of their delineating playing areas and colour properties of the camera view, and could be parsed accordingly. This led to the creation of a feature which could follow the changes in the edge information of each camera view. This feature summarised the surface of Radon space within a single value representation. Statistical colour and shape moments were also extracted to aid in the classification of camera views. A hidden Markov model (HMM) was chosen as a suitable framework for modelling the stochastic nature of the

view features. Using maximum likelihood classification, the HMM was employed to classify the various view types. Results presented in chapter 6 show it to be effective for this purpose.

Having classified the views, chapters 4 and 5 were conceived by the need for additional features for high-level event classification. Object tracking was achieved using a colour based particle filter based on the CONDENSATION algorithm. Extensions to this technique were proposed to encourage better tracking in a constrained sports environment such as tennis and snooker. The tracker made use of the *a-priori* detected scene geometry and its relation to the real world geometry of the delineated playing area to automatically scale the candidate regions. This both increased the efficiency of the tracking algorithm while also providing better track of the object. Since the playing area surface is also known *a-priori*, a likelihood ratio of object colour to background colour was used. Tracking using this method was shown to give better tracking fidelity because it promotes tracking of the selected object and not regions which contain a large number of playing area pixels, forcing the particles to be more centred on the object to be tracked. Parzen windows were used to help track smaller objects because these objects typically do not contain enough data to empirically yield a useful histogram.

In view of the track borne out by the object, it was identified that the spatio-temporal behaviour of the object over a particular duration could be related to semantic events. It is believed that this was the first attempt at analysing this behaviour to elicit high-level semantics in the sports domain. The spatio-temporal behaviour of the objects was modelled using a HMM. It enabled six common high-level events in snooker and five tennis events to be detected. This thesis only observed the behaviour of a single object under the HMM and used supplemental features such as collisions in snooker and initial player positions in tennis to aid in the classification. As there can be many more moving objects in view, modelling the behaviour of each one could provide access to additional semantics. This could be achieved for example by modelling the behaviour of both the top and bottom tennis players with separate HMMs. This would require the calculation of a correlation measure between the two models to correctly infer the event. Alternatively a single HMM which models the behaviour of both players could be used. Significant drawbacks of this approach would include both computational expense and a need for a great deal of training material.

7.1 Future work

The vast quantity and range of archived sports footage permits this area of research to be explored for many years to come. While this thesis has dealt with the detection of semantic events for specific sports, the problem of creating a complete system for generic sports parsing and high-level event extraction remains open. A hierarchical approach is clearly the most appropriate way to engage this undertaking. A typical domain hierarchy could be structured as follows:

Level 1: Low-level content analysis :

The lowest level in the hierarchy could involve discriminating the various sports based on low-level features such as colour, motion, geometry and audio and classify them as being either indoor or outdoor, the playing surface type (grass, sand, clay, *etc.*) and an approximation of the number of participants and their motion properties. Further refinement of low-level features could detect the actual sport being played.

Level 2: Semantic Content :

High-level semantic events could then be detected by monitoring the evolution of features for specific sport domains as was presented in this thesis.

Level 3: Affective Content :

Relating the resulting emotions from semantic events is key in further enhancing the retrieval for the user.

Possible improvements to the current techniques implemented are discussed under the headings of the levels in the domain hierarchy.

Low-level content analysis

Improvements in spatial and temporal segmentation are essential for this stage in the hierarchy. To this end, a method based on estimating the unknown standard deviation of a centred normal distribution from a mixture density is currently being researched for applications including spatial segmentation [32]. This method has thus far shown promising results for playing area segmentation without the need for any sort of thresholding.

Spatial segmentation could also include analysis of light and shadows to deduce the playing environment. Dominant colour ratios, models for surface type and the geometry of the scene could help classify the sport genre being played. Superior domain independent temporal boundary detection techniques are also required to better parse sports sequences.

Being able to distinguish between different types of broadcast sports requires sufficient domain knowledge. It has been shown that different sports footage exhibit varying global motion patterns [159] and colour content [24, 48]. Detection of slow-motion sections [47, 112] and view type could allow the sport type to be inferred.

Semantic Content

Most of the work undertaken in the thesis has dealt with this particular level in the hierarchy. While both visual and audio features were exploited to detect semantic events, they were applied separately. A future improvement could be to use both streams under the same probabilistic framework. HMMs allow for multimodal integration so it makes sense to exploit this powerful trait of the model.

Both audio and text have been shown to be valuable for detecting semantic events in sports footage [2, 31, 91, 97]. Crowd cheers can signify that a score has been made or a significant event has occurred and since the score is normally on screen at all times, OCR techniques should be used to supplement the existing audio visual data to identify which player or team has scored.

While the global view captures most of the important events, the same events can on occasion, be captured from other angles. Assuming that a stream of the full view is being captured, which is increasingly the case with DTV, this becomes a moot observation and all processing can be carried out on the required global view. In any case, other views must be considered to embody additional useful information which can be parsed for further high-level events. For example, in soccer, the global view would be that of an elevated side on view of the pitch. Such a view might not capture an off the ball incident. However, another camera could allow such an event to be detected.

There has been some research recently in attempting to resolve annotating keywords with visual features in images [100]. Fusing the existing keywords with segmented objects in this way could provide access to additional semantics.

Affective Content

As was discussed in section 2.3, high-level content comes in two flavours; semantic content and affective content. The purpose of this work was to extract the semantic events from the footage, for indexing and summarisation purposes. The occurrence of these events can be exploited to gauge the level of excitement of a sport, thereby accessing the affective content. For example, if a user wished to find all the exciting games in a set of tennis, this might be signified by a high presence of base-line rallies, several aces and many points exchanged around the deuce/advantage period of the game. A tedious game might be a result of a succession of aces or double faults. This would require supplemental feature extraction such as temporal information and character recognition to recognise to whom the points were awarded. A hierarchical HMM such as that used by Cohen et al [26] for emotion recognition could exploit the state sequence generated by the event detection HMMs to drive a HMM which recognises exciting, boring or eventful games. The degree or quality of affective content could be given further discriminating power by monitoring the frequency of the semantic events.

In any system such as this, user feedback is essential for improving the retrieval. While relevance feedback is a relatively mature field of research for text and low-level content based image retrieval, it could prove to be a promising research area which would enhance the retrieval of high-level sports events in video. To a certain extent the training methods for high-level events provided by this thesis are a form of relevant feedback. If the user is not happy with the retrieval provided by the system, they can tailor the retrieval for their own viewing purposes.

Continuing work in sports retrieval in Sigmedia

A front end browser is currently being produced by the Sigmedia group in Trinity College Dublin. The browser allows the user to search through the footage by the labelled events. It takes as input the time indexed footage with an event label and displays the event clip. A content aware summary of the event is generated using the methods outlined in section 6.6. The first detected content aware frame is used as a keyframe to select the event.

By detecting events and the player/team to which the points are awarded, the burden of human editing of sports footage could be alleviated. Using the detected event type along with view classification, the HMM could be used as a generative model which has been trained with established editing techniques. Robotic cameras which capture the global view could also be used to track the required objects.

The fruits of the collaboration between members of Sigmedia for sports retrieval are to be presented on Scope [133], a television program on RTE (the national Irish broadcaster) which encourages teenagers to pursue Engineering and Science courses.

It is hoped that this thesis has illustrated that with the correct combination of classification and modelling tools and exploiting the user context, high-level CBVR is feasible.

A

Hidden Markov Models

A HMM can be defined by a set of three parameters, and the known model specifications K and N . If this full set is known the HMM can be used either as a generative model or to compute how the output sequence was generated.

The elements of a general discrete HMM are:

- A set of states $S = \{S_1, S_2, \dots, S_N\}$. The state at time t is q_t : Process moves from one state to another in a Markovian fashion.
- Matrix of transition probabilities $\mathbf{A} = (a_{ij})$ where $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$, $1 \leq i, j \leq N$. It defines probabilistically how the process moves among states, where a_{ij} obeys the standard constraints:

$$\begin{aligned} a_{ij} &\geq 0, & 1 \leq i, j \leq N \\ \sum_{j=1}^N a_{ij} &= 1, & 1 \leq i \leq N \end{aligned} \tag{A.1}$$

- Set of discrete quantised observations $V = \{x_1, x_2, \dots, x_K\}$. In a given state, observations are generated according to a distribution described by \mathbf{B} , discussed next.
- Matrix of observation probabilities $\mathbf{B} = b_j(x_k)$ where $b_j(x_k) = P(X_t = x_k | q_t = S_j)$, $1 \leq j \leq N$, $1 \leq k \leq K$: It defines the pdf of observations given the state. In the discrete case \mathbf{B} obeys the constraints:

$$\begin{aligned} b_j(x_k) &\geq 0, & 1 \leq j \leq N \\ \sum_{k=1}^K b_j(x_k) &= 1, & 1 \leq j \leq N \end{aligned} \tag{A.2}$$

In essence, each state has a probability mass function associated with it that dictates how likely a particular observation is from that state.

- Vector of initial probabilities $\pi = \{\pi_i\}$; $\pi_i = P(q_1 = S_i)$, $1 \leq i \leq N$: It defines the probability of the initial state.

So, given K and N a succinct definition of a HMM can be given by λ , where

$$\lambda = \{\mathbf{A}, \mathbf{B}, \pi\} \quad (\text{A.3})$$

Given this definition of a HMM, there are three central issues of evaluation, decoding and estimation which have to be resolved before a HMM can be applied to a specific problem.

Issue 1: Evaluating the observation sequence likelihood. Given the quantised observation sequence $\mathbf{X} = \{X_1, X_2, \dots, X_T\}$, and the model $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, find the probability that the HMM actually generated the sequence regardless of the particular state sequence. This can then be used as a sort of ranking system where there are a number of competing HMM classes and the one that best fits the observation is to be found.

Issue 2: Calculation of the optimal state sequence and the individually most likely state. Given the quantised observation sequence $\mathbf{X} = \{X_1, X_2, \dots, X_T\}$, and the model $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, find the most likely sequence of states, $\mathcal{S} = \{q_1, q_2, \dots, q_T\}$, that produced the observation sequence.

Issue 3: HMM parameter estimation using Baum-Welch. Given the quantised observation sequence $\mathbf{X} = \{X_1, X_2, \dots, X_T\}$, calculate the model that best fits the observation data. (*i.e.* evaluate the model, $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, that maximises the likelihood $P(\mathbf{X}|\lambda)$).

The following sections outline the derivations and solutions to each of each issues.

A.1 Issue 1: Evaluating the observation likelihood

Evaluating the observation likelihood, $P(\mathbf{X}|\lambda)$, can be solved directly by marginalising the joint probability $P(\mathbf{X}, \mathcal{S}|\lambda)$, where $\mathcal{S} = \{q_1, q_2, \dots, q_T\}$.

$$P(\mathbf{X}|\lambda) = \int P(\mathbf{X}, \mathcal{S}|\lambda) d\mathcal{S} \quad (\text{A.4})$$

Assuming that the observation sequence \mathbf{X} has length T :

$$P(\mathbf{X}|\lambda) = \int P(\mathbf{X}, \mathcal{S}|\lambda) d\mathcal{S} \quad (\text{A.5})$$

$$P(\mathbf{X}|\mathcal{S}, \lambda) = b_{q_1}(X_1)b_{q_2}(X_2)\dots b_{q_T}(X_T)$$

The probability that this state sequence results is

$$P(\mathcal{S}|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \dots a_{q_{T-1} q_T} \quad (\text{A.6})$$

The joint probability of \mathbf{X} and \mathcal{S} is therefore

$$P(\mathbf{X}, \mathcal{S}|\lambda) = P(\mathbf{X}|\mathcal{S}, \lambda)P(\mathcal{S}|\lambda) \quad (\text{A.7})$$

Summing over all state sequences results in

$$P(\mathbf{X}|\lambda) = \sum_{\mathcal{S}} P(\mathbf{X}|\mathcal{S}, \lambda)P(\mathcal{S}|\lambda) \quad (\text{A.8})$$

This, however, results in an exponentially increasing complexity over time of $\mathcal{O}(2TN^T)$, as the sum is taken over all possible routes through the trellis. The calculation is computationally unfeasible even for a HMM with a small number of states, N , and a short observation sequence, T .

A more elegant solution to the problem of finding the observation probability is to consider the HMM in trellis representation, or alternatively, as a finite state machine (FSM) extended over time (Figure A.1). This allows an algorithm called the ‘forward-backward algorithm’ [124] to be implemented where a variable is declared that can be calculated inductively at each time step.

The basic function of the forward-backward algorithm is to efficiently propagate state information through the trellis. The following sections detail the derivations of the forward part of the algorithm.

A.1.1 Derivation of the forward variable.

For each state in the trellis the forward variable, $\alpha_t(i)$, is defined as the joint probability of observing the partial observation sequence $\mathbf{X} = \{X_1, X_2, \dots, X_t\}$ and being in state S_i at time t :

$$\alpha_t(i) = P(X_1, X_2, \dots, X_t, q_t = S_i) \quad (\text{A.9})$$

The forward variable is calculated recursively starting at the left hand side of the trellis and working to the right as illustrated in figure A.2.

1. Initialisation of forward variable for $1 \leq i \leq N$.

$$\begin{aligned} \alpha_1(i) &= P(X_1, q_1 = S_i) \\ &= P(X_1|q_1 = S_i)P(q_1 = S_i) \\ \alpha_1(i) &= b_i(X_1) \pi_i \end{aligned}$$

2. Derivation of induction step for $1 \leq i \leq N$ and $1 \leq t \leq T - 1$.

$$\begin{aligned} \alpha_{t+1}(j) &= P(X_1, \dots, X_{t+1}, q_{t+1} = S_j) \\ &= \sum_{i=1}^N P(X_1, \dots, X_{t+1}, q_t = S_i, q_{t+1} = S_j) \\ &= \sum_{i=1}^N P(X_{t+1}, q_{t+1} = S_j | X_1 \dots X_t, q_t = S_i) P(X_1 \dots X_t, q_t = S_i) \\ &= \sum_{i=1}^N P(X_{t+1}, q_{t+1} = S_j | q_t = S_i) \alpha_t(i) \\ &= \sum_{i=1}^N P(X_{t+1} | q_{t+1} = S_j) P(q_{t+1} = S_j | q_t = S_i) \alpha_t(i) \\ \alpha_{t+1}(j) &= \sum_{i=1}^N (a_{ij} \alpha_t(i)) b_j(X_{t+1}) \end{aligned}$$

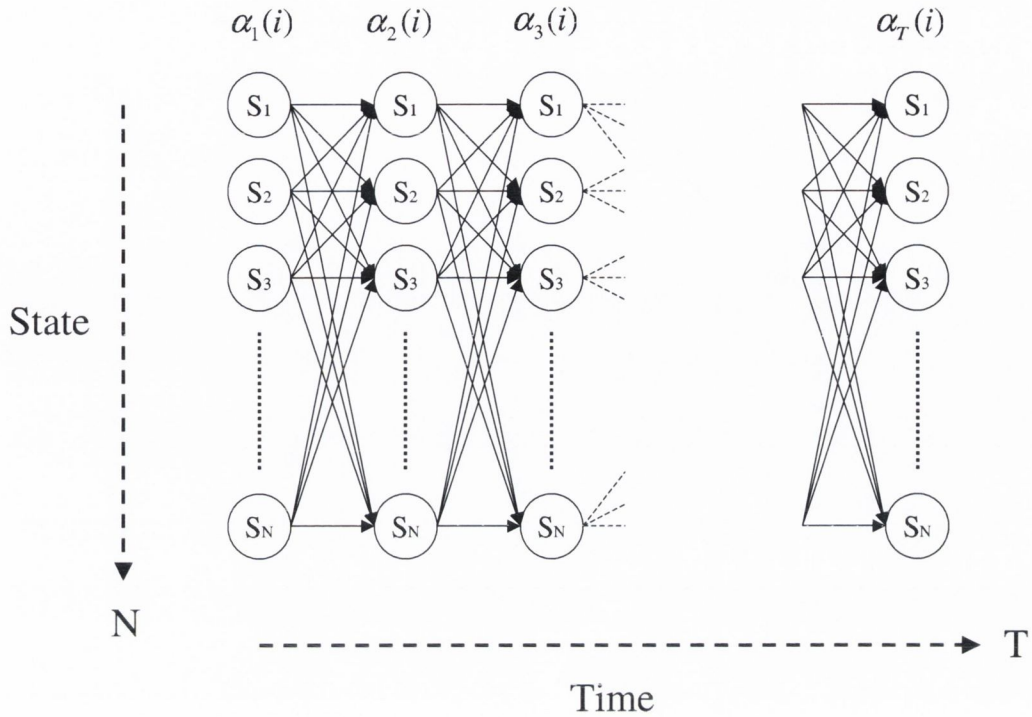


Figure A.1: Trellis representation of HMM with N states and trellis section repeated $T = 3$ times. Transitions between states defined by matrix \mathbf{A} and observation generation defined by matrix \mathbf{B}

3. Termination.

$$\begin{aligned}
 P(\mathbf{X}|\lambda) &= \sum_{i=1}^N P(\mathbf{X}|q_T = S_i, \lambda)P(q_T = S_i|\lambda) \\
 &= \sum_{i=1}^N P(\mathbf{X}, q_T = S_i|\lambda) \\
 P(\mathbf{X}|\lambda) &= \sum_{i=1}^N \alpha_T(i)
 \end{aligned}$$

This is a much more computationally efficient method than the ‘direct method’, being of $\mathcal{O}(TN^2)$.

A.2 Issue 2: Calculation of optimal state sequence and most likely state

The state sequence can be calculated in two ways. The first (section A.2.1) exploits the forward and backward variables to calculate a locally optimal state. It does so by selecting the state which maximises the probability of that state occurring at time t given all observations. The second uses the Viterbi algorithm (section A.2.4) to calculate an optimal state sequence by selecting a state from which the path to the current state is most likely.

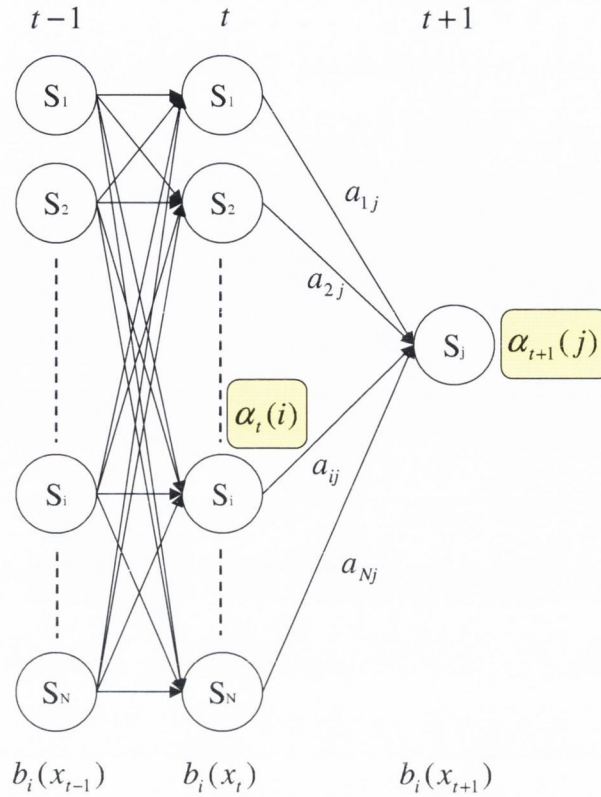


Figure A.2: Illustration of operations to obtain $\alpha_{t+1}(j)$.

A.2.1 Locally optimal state selection

This method maximises the expected number of correct states, therefore generating a locally optimised path. The decoding algorithm determines the state which should be arrived in by calculating the highest probability of all paths coming from the previous step and producing the desired observation. This is a local optimisation, therefore the algorithm does not guarantee that the path is actually allowable. (*e.g.* the probability of a certain state at a time step t may be the maximum at that instance according to the calculations, but the path may not be valid as a null transition in a_{ij} may be given).

The state path is produced by defining a variable $\gamma_t(i)$ as the probability of being in state S_i at time t given the observation sequence $\mathbf{X} = \{X_1 X_2 \dots X_T\}$. The most likely state at any time t is the one which maximizes expression A.10.

$$\gamma_t(i) = P(q_t = S_i | X_1 \dots X_T), \quad 1 \leq t \leq T \tag{A.10}$$

Forward recursion has been presented, now backward recursion is shown and how γ is

calculated from α and β (the backward variable).

A.2.2 Derivation of the backward variable

For each state in the trellis the backward variable, $\beta_t(i)$, is defined as the conditional probability of observing the partial observation sequence $\{X_{t+1}, X_{t+2}, \dots, X_T\}$ given that state S_i had been occupied at time t .

$$\beta_t(i) = P(X_{t+1}, X_{t+2}, \dots, X_T | q_t = S_i) \tag{A.11}$$

Similar to the calculation of the forward variable, the backward variable is calculated recursively, but in this case beginning at the right hand side of the trellis and working to the left. An illustration of the operations needed to obtain the backward variable is given in figure A.3.

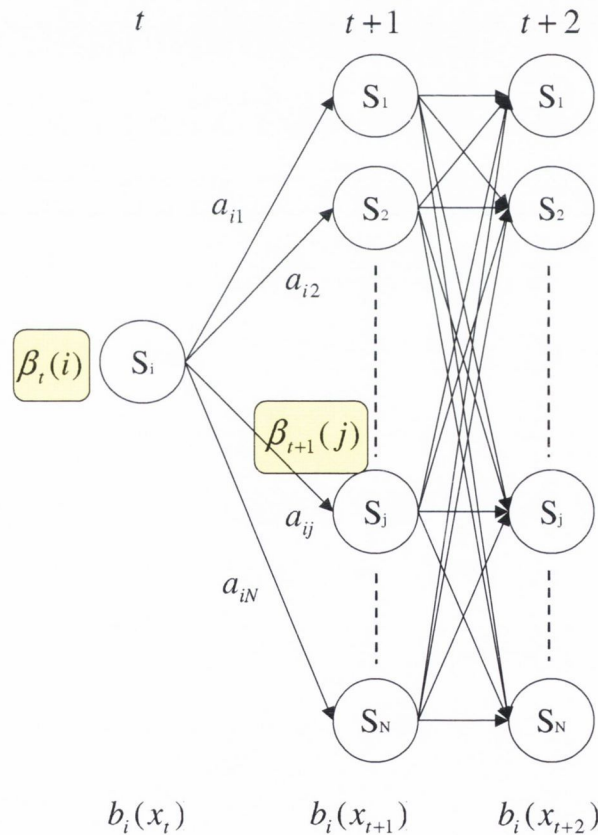


Figure A.3: Illustration of operations to obtain $\beta_{t+1}(j)$ for one time step.

1. Initialisation of backward variable at time $t = T$ and for states $1 \leq i \leq N$.

$$\beta_T(i) = 1, \quad \text{where } 1 \leq i \leq N \quad (\text{A.12})$$

2. Derivation of the induction step for $2 \leq t \leq T - 1$ and $1 \leq i \leq N$.

$$\begin{aligned} \beta_t(i) &= P(X_{t+1}, \dots, X_T | q_t = S_i) \\ &= \sum_{j=1}^N P(X_{t+1}, \dots, X_T, q_{t+1} = S_j | q_t = S_i,) \\ &= \sum_{j=1}^N P(X_{t+1}, q_{t+1} = S_j | X_{t+2}, \dots, X_T, q_t = S_i) \beta_{t+1}(j) \\ &= \sum_{j=1}^N P(X_{t+1} | q_{t+1} = S_j) P(q_{t+1} = S_j | q_t = S_i) \beta_{t+1}(j) \\ \beta_t(i) &= \sum_{j=1}^N b_j(X_{t+1}) a_{ij} \beta_{t+1}(j) \end{aligned}$$

3. Termination.

$$\beta_1(1) = \sum_{j=1}^N b_j(X_2) a_{1j} \beta_2(j) \quad (\text{A.13})$$

A.2.3 Derivation of the optimal state at time t .

$\gamma_t(i)$ is a variable which is the probability of being in state S_i at time t given the entire observation sequence. $\gamma_t(i)$ can be calculated using the forward and backward variables where γ has already been defined as the probability of being in state S_i at time t given the observation sequence $\mathbf{X} = \{X_1 X_2 \dots X_T\}$.

$$\begin{aligned} \gamma_t(i) &= P(q_t = S_i | X_1 \dots X_T) \\ &= \frac{P(X_1 \dots X_T, q_t = S_i)}{P(X_1 \dots X_T)} \\ &= \frac{P(X_1 \dots X_T | q_t = S_i) P(q_t = S_i)}{P(X_1 \dots X_T)} \end{aligned}$$

Since $(X_1 \dots X_t)$ and $(X_{t+1} \dots X_T)$ are independent given q_t the probability of the entire observation sequence can be re-written as:

$$P(X_1 \dots X_t, X_{t+1} \dots X_T | q_t = S_i) = P(X_1 \dots X_t | q_t = S_i) P(X_{t+1} \dots X_T | q_t = S_i) \quad (\text{A.14})$$

$$\begin{aligned} \gamma_t(i) &= \frac{P(X_1 \dots X_t | q_t = S_i) P(X_{t+1} \dots X_T | q_t = S_i) P(q_t = S_i)}{P(X_1 \dots X_T)} \\ &= \frac{P(X_1 \dots X_t, q_t = S_i) P(X_{t+1} \dots X_T | q_t = S_i)}{P(X_1 \dots X_T)} \\ \gamma_t(i) &= \frac{\alpha_t(i) \beta_t(i)}{\sum_{i=1}^N \alpha_t(i) \beta_t(i)} \end{aligned}$$

Where,

$$P(X_1 \dots X_T) = \sum_{i=1}^N P(X_1 \dots X_T, q_t = S_i) = \sum_{i=1}^N \alpha_t(i) \beta_t(i) \quad (\text{A.15})$$

The most likely state q_t at time t is the one which maximises $\gamma_t(i)$.

$$q_t = \arg \max_{1 \leq i \leq N} [\gamma_t(i)], \quad 1 \leq t \leq T \quad (\text{A.16})$$

A.2.4 Viterbi algorithm: Most likely path via Dynamic programming

The Viterbi algorithm is used to obtain the most likely state sequence $\mathcal{S} = \{q_1, q_2, \dots, q_T\}$ (or the sequence which maximises $P(\mathcal{S}|X_1, X_2, \dots, X_T)$ given the observation vector. It is based on the Dynamic Programming principles proposed by Bellman [8]. The evaluation is similar to that of the forward algorithm except at each state, instead of summing the probabilities from previous states, the maximum is found so there exists only the most probable path to the next state. The computation begins at time $t = 1$ and continues to the right, as with the forward algorithm, to $t = T$.

In order to keep track of the most likely state sequence, a state pointer variable, $\phi_t(i)$, is defined for each state and timestep. This pointer contains the argument which maximises $\delta_t(i)a_{ij}$ and is used to find the optimal path through the trellis. The most likely state q_t^* , or the individually most likely state at any time t , is calculated by finding the argument which maximises $\delta_t(i)$. The three steps involved in calculating the optimal state sequence are similar to the derivation of α_T with the $\sum_{i=1}^N$ being replaced with a $\max_{i=1}^N$.

1. Initialisation: As with the forward variable, computation begins at $t = 1$ for states $1 \leq i \leq N$. The state reference pointer is initialised at 0.

$$\begin{aligned}\delta_1(i) &= \pi_i b_i(X_1) \\ \phi_1(i) &= 0\end{aligned}\tag{A.17}$$

2. Induction Step: For each state in the trellis the parameter $\delta_t(i)$ is defined as the maximum of the joint probability of the partial observation sequence $\mathbf{X} = \{X_1, X_2, \dots, X_t\}$ and the state sequence. To retrieve the state sequence, the argument which maximises the previous equation must be kept track of (for $2 \leq t \leq T - 1$ and $1 \leq j \leq N$).

$$\begin{aligned}\delta_t(j) &= P(X_1, \dots, X_t, q_t = S_j) \\ &= \max_{1 \leq i \leq N} [P(X_1, \dots, X_{t-1}, q_{t-1} = S_i, q_t = S_j)] \\ &= \max_{1 \leq i \leq N} [P(X_t, q_t = S_j | X_1, \dots, X_{t-1}, q_{t-1} = S_i) P(X_1, \dots, X_{t-1}, q_{t-1} = S_i)] \\ &= \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(X_t) \\ \phi_t(j) &= \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}]\end{aligned}$$

3. Termination: At time $t = T$, the best single path is

$$\begin{aligned}P^* &= \max_{1 \leq i \leq N} [\delta_T(i)] \\ q_T^* &= \arg \max_{1 \leq i \leq N} [\delta_T(i)]\end{aligned}\tag{A.18}$$

4. Backtracking: The optimal state sequence is uncovered by backtracking through the trellis according to

$$q_t^* = \phi_{t+1}(q_{t+1}^*), \quad t = T - 1, \dots, 1\tag{A.19}$$

A.3 Issue 3: HMM parameter estimation using Baum-Welch

The goal in HMM learning is to determine the model parameters from an ensemble of feature vectors or training corpus. The most likely or optimal parameters cannot be directly solved from the data but a good estimate can be obtained using the Baum-Welch algorithm [7]. The algorithm calculates the expected values of the parameters to iteratively update the model using user supplied training. Convergence is assumed when the change in update is very small. The Baum-Welch algorithm is an instance of a generalised expectation-maximisation (GEM) algorithm.

A.3.1 Baum-Welch algorithm

Assuming a model $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, the Baum-Welch algorithm attempts to obtain a model λ' which locally maximises $P(\mathbf{X}|\lambda)$. The process is iterated until a certain threshold is met.

The model parameters, \bar{a}_{ij} , $\bar{b}_i(X_t)$, and $\bar{\pi}_i$ can be intuitively estimated as follows:

$$\bar{a}_{ij} = \frac{\text{Expected number of times a transition from } S_i \text{ to } S_j \text{ occurs}}{\text{Expected number of transitions out of state } S_i}$$

$$\bar{b}_i(X_t) = \frac{\text{Expected number of observations of } X_t \text{ while in state } S_i}{\text{Number of times in state } S_i}$$

$$\bar{\pi}_i = \text{Expected frequency in state } S_i \text{ at } t = 1$$

To describe the reestimation process, the variable $\xi_t(i, j)$ must be defined. This variable computes the probability that one hidden state follows another. (*i.e.* the probability of being in state S_i at time t and S_j at time $t + 1$ given the observation sequence $\mathbf{X} = \{X_1 X_2 \dots X_T\}$).

$$\begin{aligned} \xi_t(i, j) &= P(q_t = S_i, q_{t+1} = S_j | X_1 \dots X_T) \\ &= \frac{P(q_t = S_i, q_{t+1} = S_j, X_1 \dots X_T)}{P(X_1 \dots X_T)} \end{aligned} \quad (\text{A.20})$$

Making use of the observation independence assumption in equation A.14, and the Markov assumption equation A.20 can be rewritten as:

$$\begin{aligned} \xi_t(i, j) &= \frac{P(q_t = S_i, X_1 \dots X_t) P(X_{t+1} \dots X_T, q_{t+1} = S_j)}{P(X_1 \dots X_T)} \\ &= \frac{P(q_t = S_i, X_1 \dots X_t) P(q_{t+1} = S_j, X_{t+1}) P(X_{t+1} \dots X_T | q_{t+1} = S_j)}{P(X_1 \dots X_T)} \\ &= \frac{P(q_t = S_i, X_1 \dots X_t) P(q_{t+1} = S_j | X_{t+1}) P(X_{t+1} | q_{t+1} = S_j) P(X_{t+1} \dots X_T | q_{t+1} = S_j)}{P(X_1 \dots X_T)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(X_{t+1}) \beta_{t+1}(j)}{P(X_1 \dots X_T)} \\ \xi_t(i, j) &= \frac{\alpha_t(i) a_{ij} b_j(X_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(X_{t+1}) \beta_{t+1}(j)} \end{aligned}$$

This can also be interpreted as being the probability of going through a specific branch. Figure A.4 shows the sequences of operations required for the computation of the joint event of a system which is in S_i at time t and S_j at time $t + 1$. From figure A.4, it can be seen that $\alpha_t(i)$ accounts for the observations up until time t , a_{ij} and $b_j(x_{k+1})$ accounts for the

transition from state i to j and the resulting observation at time $t + 1$. $\beta_{t+1}(j)$ accounts for the observations from time $t + 2$ until X_T .

By summing the $\gamma_t(i)$ and $\xi_t(i, j)$ variables over times $(1 \dots T)$ and $(1 \dots T - 1)$, quantities expressing the number of times S_i is visited and transitions made from S_i , respectively can be obtained.

- Expected frequency in state S_i at $t = 1$: $\gamma_1(i)$.
- The expected number of transitions out of state S_i is: $\sum_{t=1}^{T-1} \gamma_t(i)$.
- Expected number of transitions from state S_i to S_j can be given as: $\sum_{t=1}^{T-1} \xi_t(i, j)$.
- The expected number of times in state S_i observing symbol x_k : $\sum_{t=1, X_t=x_k}^T \gamma_t(i)$. (But only times when observation is equal to x_k are counted.)

Using the expressions above, the reestimation formulae become:

$$\begin{aligned} \bar{\pi}_i &= \gamma_1(i) \\ \bar{a}_{ij} &= \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \\ \bar{b}_i(x_k) &= \frac{\sum_{t=1, X_t=x_k}^T \gamma_t(i)}{\sum_{t=1}^{T-1} \gamma_t(i)} \end{aligned} \quad (\text{A.21})$$

Based on the above procedure, by iteratively using the updated model, $\lambda' = (\bar{\mathbf{A}}, \bar{\mathbf{B}}, \bar{\pi})$, in lieu of λ and repeating the reestimation calculation, the probability of \mathbf{X} being observed can be improved until a limit governing convergence is reached.

A.3.2 Training and recognition using scaling of forward and backward variables

As the forward, $\alpha_t(i)$, and backward $\beta_t(i)$ variables are computed recursively over the length of the observation sequence, the values approach zero exponentially which, if the sequence is long enough, will be beyond the precision of most machines. In order to compensate for this, a scaling factor must be computed for all α values. β is also changed using this scale. The scaling factor c_t can be computed as:

$$\begin{aligned} c_t &= \frac{1}{\sum_{i=1}^N \alpha_t(i)}, \quad 1 \leq t \leq T \\ \tilde{\alpha}_t(i) &= c_t \alpha_t(i) \\ \tilde{\beta}_{t+1}(i) &= c_t \beta_t(i) \end{aligned} \quad (\text{A.22})$$

The scaling values do not affect the results of the re-estimated parameters as they are based on the intermediate probabilities, $\xi_t(i, j)$ and $\gamma_t(i)$ from which the scaling values in the numerator and denominator cancel.

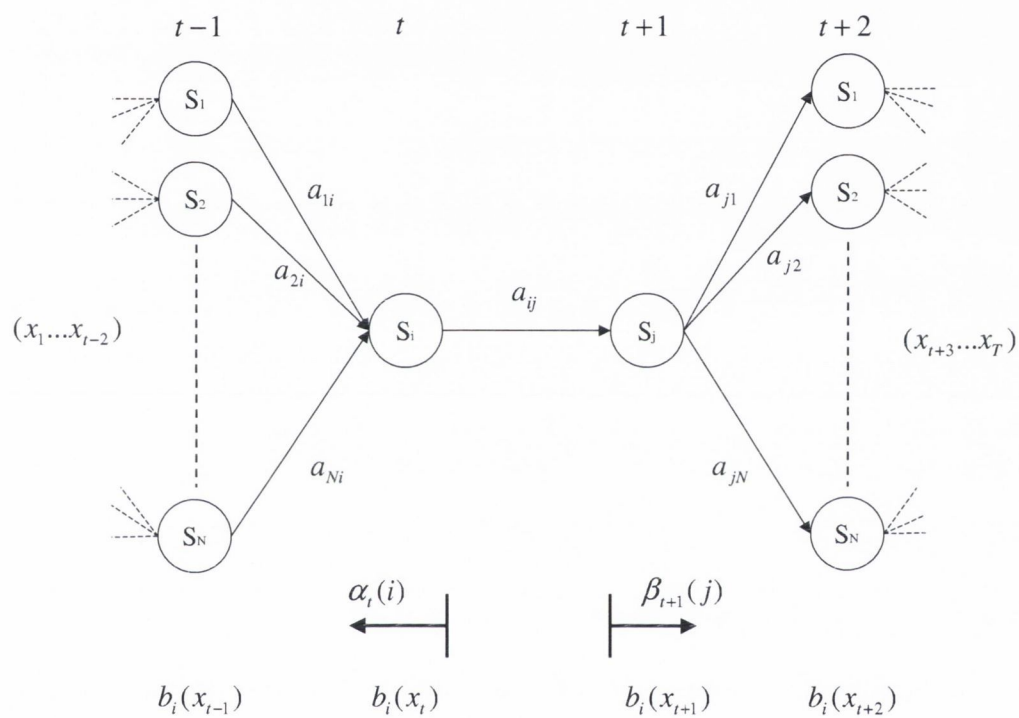


Figure A.4: Illustration of operations to obtain $\xi_k(i, j)$

B

A brief history of Snooker and Tennis, the basic rules and terminology

B.1 A brief history of Snooker

Snooker is derived from the game of billiards which had been played by aristocrats since the early 16th century. Disillusioned with the complex rules of billiards, Colonel Sir Neville Chamberlain decided on placing different coloured balls in various locations on the table and attaching certain values to these balls. The objective of this new game became one of accumulating a high score (a break) by potting the balls in a certain sequence in place of a more complicated rule set of 'winners', 'losers' and 'cannons'.

The title of the game is often attributed to Col. Chamberlain who was quoted as once saying in an article in *The Billiard Player* [34]:

'The term (snooker) was a new one to me ¹, but I soon had an opportunity of exploiting it when one of our party failed to hole a coloured ball which was close to a corner pocket. I called out to him:

"Why, you're a regular snooker." I had to explain to the company the definition of the word and, to soothe the feelings of the culprit, I added that we were all, so to speak, snookers at the game, so it would be appropriate to call the game snooker. The suggestion was adopted with enthusiasm and the game has been called snooker ever since.'

¹The nick-name 'snooker' was given to cadets at the Royal Military Academy.

A meeting was arranged in Ootacamund, India, in 1882, where the officers in Col. Chamberlain's brigade were stationed. There, precise rules were drawn up and published. However, the game was not officially recognised by the Billiards Club until the early 20th century. Since then snooker has grown into the more popular of the two games due to its more accessible rule set.

With the advent of the colour television in the 1960's, snooker emerged as the most popular table based game in the UK. Players such as Denis Taylor and Ray Reardon breathed new life into the game that up until then, was still viewed very much as a sport for the social elite. Snooker's popularity was confirmed in 1985 when 18.5 million viewers watched Dennis Taylor beat Steve Davis 18-17 in the Embassy World final at 12:23am on a Monday morning. The last frame finished on equal scores and the match was decided on a re-spotted black.

Charismatic players such as Ronnie "The Rocket" O'Sullivan and the return of Jimmy "Whirlwind" White have kept the interest in snooker at a high since a downturn in the early 1990's. Viewing figures for last years (2003) Embassy World final, televised on BBC2, in which Mark Williams withstood a comeback from Ken Doherty, before winning 18-16, peaked at 7.1 million at around 11pm and produced an audience share of 40.2%. This was the second year in a row where viewing figures exceeded 7 million for the final.

B.2 The basic rules - Snooker

Snooker is played on a green felted table measuring 1.86m in width and 3.7m in length with 21 coloured balls and one cue ball all of 52.5mm in diameter. The objective of the game is to accumulate the highest break by potting (sinking) balls in a particular order. A red must be the first ball potted followed by another ball of any colour other than red or white. The coloured balls are re-spotted after being sunk. This sequence is followed until all red balls have been potted. The player must then pot the yellow, green, brown, blue, pink and black balls in that order. These are not re-spotted. For example, to accumulate the highest break possible in snooker (147 points), each red must be potted followed in turn by the black. Having potted all reds, the remaining coloured balls are potted in the required sequence.

If the white ball is potted or if the white collides with a ball other than that declared, a foul is called. The opposing player receives 4 extra points (minimum), or the value of the ball that the white hit, and play reverts to him. Figure B.1 shows a typical snooker table with measurements, ball locations and ball values.

B.3 Snooker Terminology

Baulk: The area before the top horizontal line (the baulk line), drawn on the table. This is normally the safest area for the white ball since it should be the furthest away from the reds.

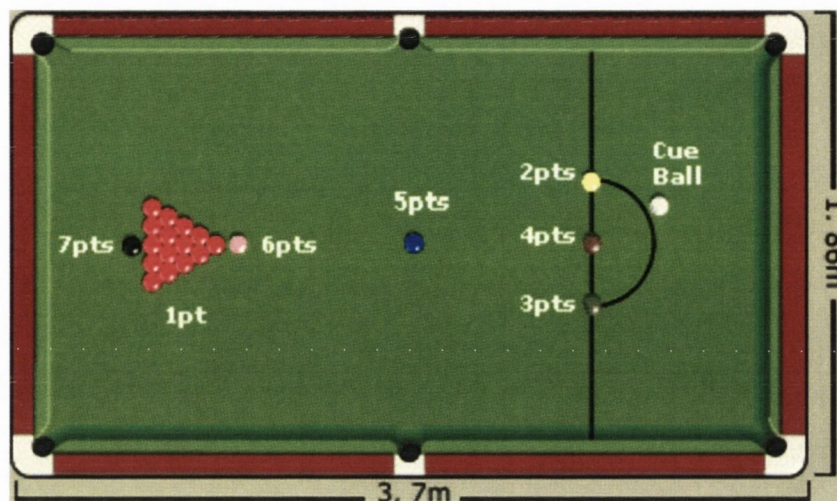


Figure B.1: A snooker table with dimensions, location of the balls and their values.

Baulk Colours: The coloured balls on the line that borders the baulk from the rest of the table (yellow, brown and green).

Break: The cumulative score of a player in one turn at the table.

Colours: Defined as the balls of the game other than the white and the reds. They all have values between two and seven as shown in figure B.1.

Cue Ball/White Ball: The white ball of the game, used to hit the other balls. The ball is hit by the cue of the player.

Cushion: The edges of the playing area are lined with cushions to allow the balls to bounce freely. A difficult shot results if the cue ball is positioned beside the cushion (on the cushion). This means that a player does not have full control over the ball.

Frame: Usually described as one “racked” game of snooker. (Not to be confused with a single image. Should be implicit in the context).

Open Table: If a player attempts a shot but misses and leaves the white ball in such an area where there is a high likelihood of balls being available to pot by his opponent.

Pocket: These are located at each corner of the snooker table and half way down the longest sides. It is not necessary for a player to call the pocket into which he is attempting to pot the ball.

Pot: When a ball rolls into a pocket.

Referee: A person that keeps track of the scores, cleans the balls, racks them and decides on fouls. He will usually have to impinge on the table to fulfil his duties.

Safety: If the player hits the cue ball so that a collision will result in a difficult shot for the opponent. A safety does not necessarily have to be a snooker (see last term). The white ball is usually hit into the baulk area or beside a cushion.

Shot-to-nothing: When the player takes on a pot while not attempting to place himself on the next ball in the sequence. These are usually long, dangerous pots and if not achieved can leave an easy opening for the opponent, but if they are, can be quite spectacular.

Snooker: Snookering an opponent is achieved by making it difficult for him to hit his next desired ball. Usually a player will try to snooker his opponent when there is no easy shot available, or when there are not enough points on the table for him to win the frame. The white ball is usually hit so that it becomes positioned behind a coloured ball that is not the next in the sequence to be hit.

B.4 The basic rules - Tennis

Tennis can be played in either indoor or outdoor environments and on a variety of court surfaces. Typically, outdoor games are played on grass, clay, rebound ace or hard courts while indoor games are played on hard court or rebound ace surfaces. Each of the four annual Grand Slam events is contested on a different surface: Grass at Wimbledon, clay at The French Open, hard court at the US Open, and rebound ace at the Australian Open. The delineating lines on all surfaces are painted white.

The type of surface will dictate the players approach to the game [22]. Grass surfaces have a low coefficient of friction and are called 'fast' surfaces. Clay on the other hand is a 'slow' surface due to its relatively high coefficient of friction. On a fast court, a serve-and-volley is an often used attack by the serving player. His opponent will find it difficult to return the serve and the point can be won by the server approaching the net and volleying the winner. Longer baseline rallies are a typical feature of harder court surfaces as the players probe for openings.

The figure below shows a schematic of an ITF (International Tennis Federation) regulation

sized tennis court.

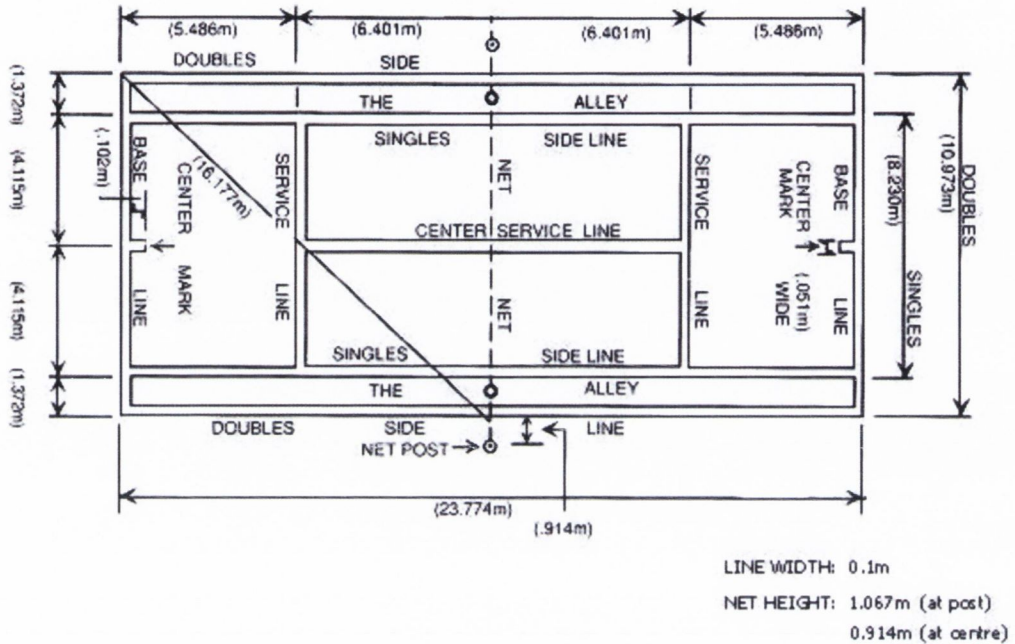


Figure B.2: A schematic of a tennis court with line dimensions.

B.5 Notes on the captured broadcast footage

The broadcast snooker and tennis footage used in this research were captured at 720×576 resolution. Each of the footage sources is from a different broadcaster and/or competition exhibiting different editing effects and camera views. The footage *Hunter* is from Sky Sports and is of the 2002 British Open between Paul Hunter and Ian McCulloch. *Higgins* shows John Higgins playing Stephen Hendry in the 2001 World Championship, broadcast on BBC while *Hendry* is the final of the 2002 Irish Open between John Higgins and Stephen Hendry shown on RTE television. Each of the footage sources was captured on different media and all are of relatively low quality. *Hendry* was recorded using VHS, *Higgins* using S-VHS and *Hunter* using a DVD copy from VHS. The low quality caused some hindrance in conducting the experiments, in particular the ball tracking. Supplemental DV footage was made available late in the research from stop.watch television. This footage was not used for evaluation of the algorithms outlined in chapters 3- 5, or view classification in chapter 6. It was however

used for event detection, the results of which are reported on in the relevant sections. The footage *King* is of the 2004 Irish Masters from RTE and was contested between Mark King and Graeme Dott.

The tennis footage used was captured on DigiBeta and consists of four separate games from 2 broadcasters. The grass court footage is from BBC coverage at Wimbledon. The first grass court game, *Pierce*, is of Mary Pierce vs Magui Serna while the second grass court match, *Hewitt* is between Lleyton Hewitt and Xavier Malisse. The final footage from Wimbledon, *Malisse*, is of Xavier Malisse against David Nalbandian. The clay court footage, *Costa*, is from RTP coverage of the Estoril Open and shows the match of Albert Costa against Todd Martin.

The system developed in this research operates on captured live broadcast footage. The camera views that are used are therefore controlled by a studio editor. Advances in digital television enables the user to interact with the video stream and select whichever camera view they would like to observe. This means that each view is continuously broadcast without interference from the others. As this thesis only considers the global view (full table for snooker and full court for tennis) for event classification, current technologies encourage and validate the use of this assumption. In section 3.2, it is shown that for non-interactive tennis and snooker streams respectively, the occurrences of these main views are quite high relative to the total duration of the clip.

C

Particle Filter

The Particle Filter is one of the principal tools in a suite of Sequential Monte Carlo (SMC) techniques [44]. It is a numerical technique for generating samples from the posterior distribution of some variable that is under scrutiny. The key point is that the technique allows the updating of the underlying sample distribution *as new data arrives*. It is different from the Kalman Filter in that it is a numerical technique that is suitable for handling non-linear systems in which the underlying distributions are non-Gaussian.

Consider that the state variable under scrutiny at time t is q_t , and it must be estimated indirectly through the observed data X_t . The essential idea of the Particle Filter is to exploit the factorisation of the posterior $p(q_t|X_t, X_{1..t-1}, q_{1..t-1})$ into a term involving a likelihood at the current time t and a prior which connects previous states with the current state: a predictive process for the states. For the work in this thesis for instance, the state to be estimated is the position of an object. Hence the likelihood encodes some image matching criterion while the predictive prior encodes the motion of the object *i.e.* how the current position relates to previous positions.

The Particle Filter therefore involves two broad stages. The first is simulation of the posterior at time $t + 1$ given observations up to time t . This is the prediction step. Secondly a weighting or “filtering” step is employed, which is a direct application of Bayes rule, to compute the posterior from observations up until time $t + 1$. This second step employs the likelihood at $t + 1$ as weights for resampling, much like importance weights would be used in importance sampling. Note that in this context the word “filtering” is not used in the traditional DSP sense. Instead it refers to the time evolution of the samples in the Particle

Filter as they are pruned or regenerated through the weighting and resampling steps.

The mean state of the system, or the posterior mean estimate, can be solved using the filtering density, $p(q_{t+1}|X_{1...t+1})$, where q_{t+1} is the state at time $t + 1$, and $X_{1...t+1}$ is the observation sequence up to time $t + 1$.

$$\mathcal{E}[p(q_{t+1}|X_{1...t+1})] = \int q_{t+1} p(q_{t+1}|X_{1...t+1}) dq_{t+1} \quad (C.1)$$

The filtering density is estimated recursively in two steps: Prediction and Update. If a first order Markovian process is assumed the transition density of the unobserved states can be described by:

$$p(q_{t+1}|q_t \dots q_1) = p(q_{t+1}|q_t) \quad (C.2)$$

and if observations are considered to be independent given a particular state the observation likelihood becomes:

$$p(X_{t+1}|q_{t+1} \dots q_1) = p(X_{t+1}|q_{t+1}) \quad (C.3)$$

In the **prediction** stage, the prior at time $t + 1$ is estimated by propagating the current filtering density, $p(q_t|X_{1...t})$, by the state transition density $p(q_{t+1}|q_t)$ according to equation C.4 below:

$$\begin{aligned} p(q_{t+1}|X_{1...t}) &= \int p(q_t, q_{t+1}|X_{1...t}) dq_t \\ &= \int p(q_{t+1}|q_t) p(q_t|X_{1...t}) dq_t \end{aligned} \quad (C.4)$$

At the next time step, $t + 1$, the filtering density is **updated** by Bayes theorem when new data is observed to arrive at the new posterior.

$$\begin{aligned} p(q_{t+1}|X_{1...t+1}) &= \frac{p(X_{1...t+1}|q_{t+1})p(q_{t+1})}{p(X_{1...t+1})} \\ &= \frac{p(X_{t+1}, X_{1...t}|q_{t+1})p(q_{t+1})}{p(X_{t+1}, X_{1...t})} \\ &= \frac{p(X_{t+1}|X_{1...t}, q_{t+1})p(X_{1...t}|q_{t+1})p(q_{t+1})}{p(X_{t+1}|X_{1...t})p(X_{1...t})} \\ &= \frac{p(X_{t+1}|X_{1...t}, q_{t+1})p(q_{t+1}|X_{1...t})p(X_{1...t})p(q_{t+1})}{p(X_{t+1}|X_{1...t})p(X_{1...t})p(q_{t+1})} \\ &= \frac{p(X_{t+1}|q_{t+1})p(q_{t+1}|X_{1...t})}{p(X_{t+1}|X_{1...t})} \end{aligned} \quad (C.5)$$

So the effective prior in equation C.5, $p(q_{t+1}|X_{1...t})$, is taken as a prediction of the posterior, $p(q_t|X_{1...t})$, in the previous time-step, given in equation C.4. However, this recursion formula cannot be implemented in practice because it can only be evaluated using high dimensional integrals [43].

An approximation to the true posterior can be achieved however by mapping the integrals of the Bayesian solution of a recursively estimated posterior to a discrete weighted sum of samples drawn indirectly from the posterior via a proposal function which is in some way related to the posterior.

Since the true posterior is not normally available to be sampled from, a proposal distribution $u(q_t|X_{1...t})$ is used in its place. From equation C.1, a general formulation of the

approximation to the posterior is:

$$\begin{aligned}
 \mathcal{E}[p(q_{t+1}|X_{1...t+1})] &= \int f(q_{t+1}) \frac{p(q_{t+1}|X_{1...t+1})}{u(q_{t+1}|X_{1...t+1})} u(q_{t+1}|X_{1...t+1}) dq_{t+1} \\
 &= \int f(q_{t+1}) \frac{p(X_{1...t+1}|q_{t+1})p(q_{t+1})}{p(X_{1...t+1})u(q_{t+1}|X_{1...t+1})} u(q_{t+1}|X_{1...t+1}) dq_{t+1} \\
 &= \frac{1}{p(X_{1...t+1})} \int f(q_{t+1}) w_{t+1} u(q_{t+1}|X_{1...t+1}) dq_{t+1}
 \end{aligned} \tag{C.6}$$

Eliminating the normalising density and where

$$w_{t+1} = \frac{p(X_{1...t+1}|q_{t+1})p(q_{t+1})}{u(q_{t+1}|X_{1...t+1})} \tag{C.7}$$

the normalised posterior becomes

$$\begin{aligned}
 \mathcal{E}[f(q_{t+1})] &= \frac{\int f(q_{t+1}) w_{t+1} u(q_{t+1}|X_{1...t+1}) dq_{t+1}}{\int p(X_{1...t+1}|q_{t+1}) p(X_{t+1}) \frac{u(q_{t+1}|X_{1...t+1})}{u(q_{t+1}|X_{1...t+1})} dq_{t+1}} \\
 &= \frac{\int f(q_{t+1}) w_{t+1} u(q_{t+1}|X_{1...t+1}) dq_{t+1}}{\int w_{t+1} u(q_{t+1}|X_{1...t+1}) dq_{t+1}} \\
 &= \frac{\mathcal{E}[w_{t+1} f(q_{t+1})]}{\mathcal{E}[w_{t+1}]}
 \end{aligned} \tag{C.8}$$

According to perfect Monte Carlo simulation [44] any expectation of the form in equation C.9 can be approximated by equation C.10.

$$\mathcal{E}[f(q_{t+1})] = \int f(q_{t+1}) p(q_{t+1}|X_{1...t+1}) dq_{t+1} \tag{C.9}$$

$$\mathcal{E}[f(q_{t+1})] \approx \frac{1}{N} \sum_{i=1}^N f(q_{t+1}^{(i)}) \tag{C.10}$$

This means that by sampling from the proposal function $u(\cdot)$, equation C.8 can be approximated as equation C.11.

$$\mathcal{E}[f(q_{t+1})] \approx \frac{\frac{1}{N} \sum_{i=1}^N w_{t+1}^{(i)} f(q_{t+1}^{(i)})}{\frac{1}{N} \sum_{i=1}^N w_{t+1}^{(i)}} \tag{C.11}$$

By replacing $f(q_{t+1})$ with q , statistics such as the MMSE or MAP estimates of the state can be calculated. A mechanism to sequentially update the weights is given in equation C.12 which is proved fully in Doucet et al [44].

$$w_{t+1} = w_t \frac{p(X_{t+1}|q_{t+1})p(q_{t+1}|q_t)}{u(q_{t+1}|q_{1...t}, X_{1...t+1})} \tag{C.12}$$

The iterative algorithm for the generic particle filter, which is implemented for the tracking problem in this thesis, is given in several papers [4, 107, 118], so is not reproduced here.

D

The Radon Transform

The Radon transform enables the value of a 2D function at an arbitrary point to be uniquely obtainable by calculating the integrals along the lines of all directions passing the point. The primary goal of the transform is to simplify the process of finding global geometrical objects (circles and lines for example) in the (x, y) image domain by re-parameterisation. The problem of finding the object is then converted into one of local peak detection. For straight line detection, the Radon transform uses the $\rho = x \cos\theta + y \sin\theta$ form of the line (illustrated in figure D.1), where ρ is the perpendicular distance of the line to the origin and θ is the angle of the perpendicular to the horizontal in image space.

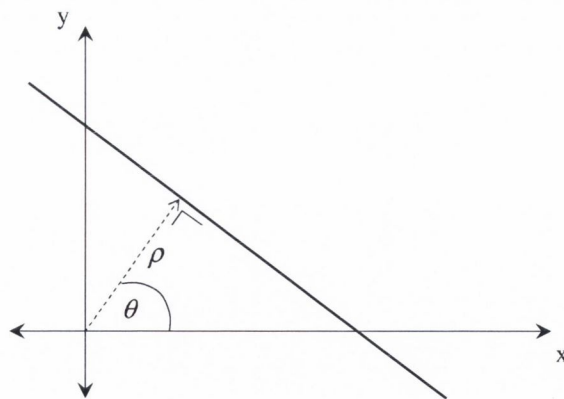


Figure D.1: (ρ, θ) representation of a straight line.

It has been shown in [36, 147] that the Radon transform of a point is a sine wave. Figure D.2 (top) shows an arbitrary point (x^*, y^*) and its corresponding Radon transform.

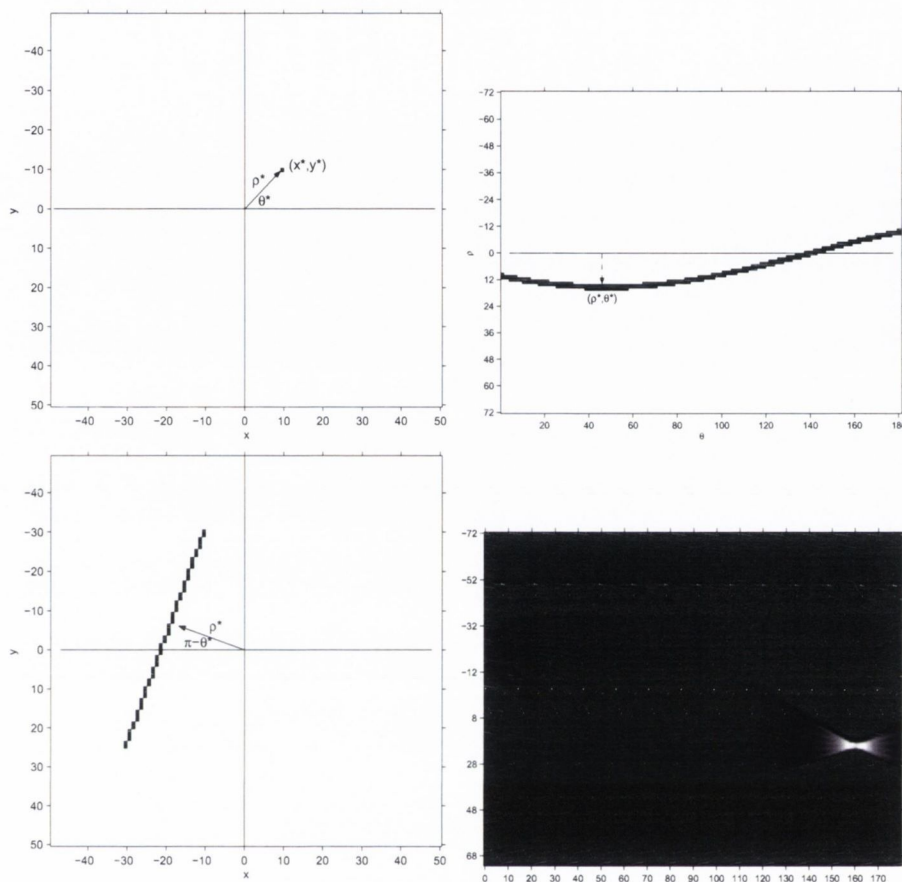


Figure D.2: Top: Radon Transform of a point (x^*, y^*) in (x, y) image space to a sine wave with parameters (ρ^*, θ^*) in (ρ, θ) Radon space. Bottom: Radon Transform of a line resulting in a peak in accumulated Radon space, where the location of the peak (ρ^*, θ^*) are the parameters of the image space line.

By applying this procedure to several points, (e.g. the discrete points in a line), where each point results in a corresponding sine wave of different (ρ, θ) parameters, integration of the waves in Radon space results in a peak where the waves intersect. The maximum in the accumulated Radon space are the (ρ, θ) parameters of the line in image space. Figure D.2 (bottom), illustrates the Radon transform of a line.

As a result of the strong geometry of some sports playing surfaces, the lines or playing surface boundaries can be extracted from the image using a combination of segmentation and edge detection.

E

Results of snooker and tennis view classification

This appendix tabulates the results from the view classification of section 6.4. The quantised statistical shape and colour features were used to train a discrete hidden Markov model (DHMM). In section E.1, tables E.1-E.7 provide recall and precision results for the classification using the features individually for snooker and tennis footage. In section E.2, tables E.8-E.14 show the result of cascading the two classifiers for the same footage.

E.1 View Classification using features individually

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	83.33 %	100 %	81.82%
Precision	75.00 %	91.30%	94.74%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	90.48 %	95.24%	100%
Precision	90.48 %	90.91%	86.96%

Table E.1: Results of the classification using shape and colour moments on the Higgins sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	92.31 %	92.31%	92.86%
Precision	60.00 %	60.00%	65.00%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	91.67 %	85.71%	100%
Precision	55.00 %	63.16%	71.43%

Table E.2: Results of the classification using shape and colour moments on the Hendry sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	80.95 %	83.64%	81.13%
Precision	89.47 %	94.85%	90.53%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	83.96 %	81.73%	86.41%
Precision	90.82 %	88.54%	88.12%

Table E.3: Results of the classification using shape and colour moments on the Hunter sequence .

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	100 %	100 %	100%
Precision	81.25 %	75.00%	62.50%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	91.67 %	100%	87.50%
Precision	73.33 %	81.25%	100%

Table E.4: Results of the classification using shape and colour moments on the Pierce sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	90.00 %	83.33%	100%
Precision	52.94%	62.50 %	72.22%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	75.00 %	71.43%	76.92%
Precision	85.71 %	71.43%	66.67%

Table E.5: Results of the classification using shape and colour moments on the Malisse sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	80.39 %	81.63%	77.36%
Precision	83.67 %	80.00%	87.23%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	70.83 %	75.51%	93.75%
Precision	75.56%	78.72%	80.36%

Table E.6: Results of the classification using shape and colour moments on the Hewitt sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	83.33 %	80.56%	93.06%
Precision	95.24 %	95.08%	95.71%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	Colour
Recall	82.43 %	86.30%	89.04
Precision	98.39 %	96.92%	97.01

Table E.7: Results of the classification using shape and colour moments on the Costa sequence .

E.2 View Classification by cascading the classifiers

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	90.91%	100%	82.61%
Precision	90.91 %	95.83%	95.00%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	91.30%	95.65%	
Precision	95.45 %	95.65%	

Table E.8: Results of the classification using a combination of colour and each of the shape features for the Higgins sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	100%	100%	95.00%
Precision	83.33 %	79.17%	82.61%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	100%	100%	
Precision	79.17 %	82.61%	

Table E.9: Results of the classification using a combination of colour and each of the shape features for the Hendry sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	81.74 %	82.61 %	82.61%
Precision	92.16 %	95.96%	94.06%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	84.35 %	82.61%	
Precision	93.27 %	92.23%	

Table E.10: Results of the classification using a combination of colour and each of the shape features for the Hunter sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	87.50%	87.50%	87.50%
Precision	100%	100%	100%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	87.50%	87.50%	
Precision	100%	100%	

Table E.11: Results of the classification using a combination of colour and each of the shape features for the Pierce sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	75.00%	75.00%	81.25%
Precision	80.00%	80.00%	86.67%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	75.00%	75.00%	
Precision	85.71 %	80.00%	

Table E.12: Results of the classification using a combination of colour and each of the shape features for the Malisse sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	90.74%	86.79%	90.91%
Precision	83.64%	83.33%	84.21%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	85.19 %	90.57%	
Precision	88.46%	88.89%	

Table E.13: Results of the classification using a combination of colour and each of the shape features for the Hewitt sequence.

Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{100}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{010}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{001}^{shape})$
Recall	82.67 %	82.67%	91.89%
Precision	100%	100%	98.55%
Moments	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{011}^{shape})$	$(\mathcal{M}_{002}^{shape}, \mathcal{M}_{101}^{shape})$	
Recall	81.33 %	85.33%	
Precision	100%	100%	

Table E.14: Results of the classification using a combination of colour and each of the shape features for the Costa sequence.

Bibliography

- [1] Y. Abdeljaoued, T. Ebrahimi, C. Christopoulos, and I. M. Ivars. A new algorithm for shot boundary detection. In *10th European Signal Processing Conference (EUSIPCO 00)*, pages 151–154, Tampere, Finland, September 2000.
- [2] N. Adami, R. Leonardi, and P. Migliorati. An overview of multi-modal techniques for the characterization of sport programmes. In *Visual Communications and Image Processing (VCIP'03)*, pages 1296–1306, University of Italian Switzerland (USI), Lugano, Switzerland, July 2003.
- [3] W. H. Adams, G. Iyengar, C.-Y. Lin, M. R. Naphade, C. Neti, H. J. Nock, and J. R. Smith. Semantic indexing of multimedia content using visual, audio, and text cues. *EURASIP Journal on Applied Signal Processing*, 2:1–16, 2003.
- [4] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50:174–188, February 2002.
- [5] J. Assfalg, M. Bertini, A. Del Bimbo, W. Nunziati, and P. Pala. Soccer highlight detection and recognition using HMMs. In *IEEE International Conference on Multimedia and Expo (ICME '02)*, volume 1, pages 825–828, Swiss Federal Institute of Technology, EPFL, Lausanne, Switzerland, August 2002.
- [6] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Jain, and C. F. Shu. Virage image search engine: an open framework for image management. In *The International Society for Optical Engineering. (Storage and Retrieval for Image and Video Databases IV)*, volume 2670, pages 76–87, San Diego/La Jolla, California, USA, January 1996.
- [7] L. E. Baum. An inequality and associated maximisation technique in statistical estimation for probabilistic functions of markov processes. *Inequalities*, 3:1–8, 1972.
- [8] R. E. Bellman. *Dynamic Programming*. University Press, 1957.
- [9] M. Bertini, A. Del Bimbo, and W. Nunziati. Semantic annotation for live and posterity logging of video documents. In *Visual Communications and Image Processing*

- (*VCIP'03*), pages 1307–1316, University of Italian Switzerland (USI), Lugano, Switzerland, July 2003.
- [10] I. Biederman. Recognition-by-components: a theory of human image understanding. *Psychological Review*, 94:115–147, 1987.
- [11] C. Biernacki, G. Celeux, and G. Govaert. Assessing a mixture model for clustering with the Integrated Completed Likelihood. *IEEE Transactions on PAMI*, 22(3):719–725, 2000.
- [12] S.M. Bileschi and B. Heisele. Advances in component based face detection. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures (AMFG 2003)*, pages 149–156, Nice, France, October 2003.
- [13] J. A. Bilmes. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian Mixture and Hidden Markov Models. Technical Report TR-97-021, International Computer Science Institute, 1998.
- [14] A. Del Bimbo. *Visual Information Retrieval*. Morgan Kaufmann, 1999.
- [15] A. Del Bimbo and P. Pala. Visual image retrieval by elastic matching of user sketches. *Pattern Analysis and Machine Intelligence*, 19:121–132, 1997.
- [16] J Black, K. Kahol, P. Tripathi, P. Kuchi, and S. Panchanathan. Indexing natural images for retrieval based on Kansei factors. In *Human Vision and Electronic Imaging (HVEI) Conference*, volume 5292, pages 363–375, San Jose, California, USA, June 2004.
- [17] A. Blake and M. Isard. *Active Contours*. Springer, 1998.
- [18] A.F. Bobick and J.W. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3), March 2001.
- [19] J. Boreczky, A. Girgensohn, G. Golovchinsky, and S. Uchihashi. An interactive comic book presentation for exploring video. In *ACM Conference on Human Factors in Computing Systems (CHI 2000)*, pages 185–192, The Hague, The Netherlands, April 2000.
- [20] P. Bouthemy, 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.
- [21] G. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*, Q2 1998. URL: http://developer.intel.com/technology/itj/q21998/articles/art_2.htm.

- [22] H. Brody. Bounce of a tennis ball. *Journal of Science and Medicine in Sport*, 1:113–119, 2003.
- [23] L. Chaisorn, T.-S Chua, and C.-H.Lee. The segmentation of news video into story units. In *IEEE International Conference on Multimedia and Expo (ICME 02)*, volume 1, pages 73–76, Swiss Federal Institute of Technology, EPFL, Lausanne, Switzerland, August 2002.
- [24] P. Chang, M. Han, and Y. Gong. Extract highlights from baseball game video with Hidden Markov Models. In *Proceedings of the International Conference on Image Processing (ICIP '02)*, pages 609–612, Rochester, New York, USA, September 2002.
- [25] S-F. Chang. The holy grail of content-based media analysis. *IEEE Multimedia*, 9(2):6–10, April-June 2002.
- [26] I. Cohen, A. Garg, and T. S. Huang. Emotion recognition from facial expressions using multilevel HMM. In *Neural Information Processing Systems*, Denver, Colorado, USA, November 2000.
- [27] C. Colombo, A. Del Bimbo, and P. Pala. Semantics in visual information retrieval. *IEEE Multimedia*, 3:38–53, 1999.
- [28] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using Mean Shift. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 142–149, Hilton Head Island, South Carolina, USA, June 2000.
- [29] A. Crawford. CASMS browser. Technical report, Trinity College Dublin, Ireland, 2004. URL: http://www.mee.tcd.ie/~sigmedia/research/indexing/sport_indexing.php.
- [30] R. Dahyot, P. Charbonnier, and F. Heitz. Unsupervised statistical change detection in camera-in-motion video. In *IEEE Proceedings of the International Conference on Image Processing*, volume 1, pages 638–641, Thessaloniki, Greece, October 2001.
- [31] R. Dahyot, A. C. Kokaram, N. Rea, and H. Denman. Joint audio visual retrieval for tennis broadcasts. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)*, volume 3, pages 561–564, Hong Kong, April 2003.
- [32] R. Dahyot, N. Rea, A. Kokaram, and N. Kingsbury. Inlier modeling for multimedia data analysis. In *IEEE International Workshop on Multimedia Signal Processing*, pages 482–485, Siena, Italy, September 2004.
- [33] R. Dahyot, N. Rea, and A. C. Kokaram. Sport video shot segmentation and classification. In *Visual Communications and Image Processing (VCIP 2003)*, volume 5150, pages 404–413, University of Italian Switzerland (USI), Lugano, Switzerland, July 2003.

- [34] J. Davis. The breaks came my way, 1976. URL: <http://www.eaba.co.uk/books/davis/chapter1.html>.
- [35] Institut de Recherche en Communications et en Cyberntique de Nantes. URL: <http://www.irccyn.ec-nantes.fr/irccyn/d/en/equipes/ImagesVideo/axes/moj>.
- [36] S. R. Deans. *The Radon Transform and some of its Applications*. John Wiley and Sons, 1983.
- [37] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, pages 1–38, 1977.
- [38] H. Denman and A. Kokaram. A multiscale approach to shot change detection. In *Irish Machine Vision and Image Processing (IMVIP 04)*, pages 19–25, Dublin, Ireland, September 2004.
- [39] H. Denman, N. Rea, and A. Kokaram. Content based analysis for video from snooker broadcasts. In *International Conference on Image and Video Retrieval (CIVR'02)*, volume 2383, pages 186–193, London, UK, July 2002.
- [40] H. Denman, N. Rea, and A. C. Kokaram. Content based analysis for video from snooker broadcasts. *Journal of Computer Vision and Image Understanding (CVIU): Special Issue on Video Retrieval and Summarization*, 92(2–3):141–306, November–December 2003.
- [41] C. Djeraba. Content-based multimedia indexing and retrieval. *IEEE Multimedia*, 9(2):52–60, 2002.
- [42] M. R. Dobie and P. H. Lewis. Object tracking in multimedia systems. In *IEEE International Conference on Image Processing and its Applications*, pages 41–44, Singapore, September 1992.
- [43] A. Doucet. On sequential simulation-based methods for Bayesian filtering. Technical Report CUED/F-INFENG/TR.310, University of Cambridge, 1998.
- [44] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Statistics for Engineering and Information Science. Springer, 2000.
- [45] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, 2001.
- [46] S. Eickeler and S. Muller. Content-based video indexing of tv broadcast news using Hidden Markov Models. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 1999)*, volume 6, pages 2997–3000, Phoenix, Arizona, USA, March 1999.

- [47] A. Ekin. *Sports Video Processing for Description, Summarization, and Search*. PhD thesis, Department of Electrical and Computer Engineering, The College School of Engineering and Applied Sciences University of Rochester, Rochester, New York, 2003.
- [48] A. Ekin and A. M. Tekalp. Automatic soccer video analysis and summarization. In *International Conference on Electronic Imaging: Storage and Retrieval for Media Databases*, volume 5021, pages 339–350, Santa Clara, California, USA, January 2003.
- [49] A. Ekin, A. M. Tekalp, and R. Methrotra. Integrated semantic-syntactic video event modeling for search and browsing. *IEEE Transactions on Multimedia*, 2004.
- [50] P. Enser and C. Sandom. Toward a comprehensive survey of the semantic gap in visual image retrieval. In *International Conference on Image and Video Retrieval (CIVR 03)*, volume 2728, pages 291–299, Urbana-Champaign, Illinois, USA, July 2003.
- [51] C. L. Epstein. *Mathematics of Medical Imaging*. Prentice Hall, 2003.
- [52] R. Fablet, P. Bouthemy, and P. Perez. Non-parametric statistical analysis of scene activity for motion-based video indexing and retrieval. Technical Report RR-4005, INRIA, 2000.
- [53] S. Fatsis and K. Pope. NFL scores nearly \$18 billion in tv rights. *The Wall Street Journal*, January 1998. URL: <http://subscribe.wsj.com/microexamples/articlefiles/NFLScoresNearly18BillionInTVRights.doc>.
- [54] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, B. Domand M. Gorkani, J. Hafner, D. Lee and D. Petkovic, D. Steele, and P. Yanker. Computer query by image and video content: the QBIC system. *IEEE Computer*, 28(9):23–32, 1995.
- [55] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, 2 edition, 1990.
- [56] T. Gevers and A. W. M. Smeulders. Color based object recognition. *Pattern Recognition*, 32:453–464, May 1999.
- [57] A. Girgensohn and J. Boreczky. Time-constrained keyframe selection technique. In *IEEE International Conference on Multimedia Systems*, volume 1, pages 756–761, Florence, Italy, June 1999.
- [58] Y. Gong, L. T. Sin, C. H. Chuan, H. Zhang, and M. Sakauchi. Automatic parsing of TV soccer programs. In *Proceedings of the International Conference on Multimedia Computing and Systems (ICMCS 95)*, pages 167–174, Washington, District of Columbia, USA, May 1995.

- [59] N.C. Haering, R.J. Qian, and M.I. Sezan. Detecting hunts in wildlife videos. In *IEEE International Conference on Multimedia Computing and Systems*, volume 1, pages 905–909, San Jose, California, USA, June 1999.
- [60] A. Hanjalic. Generic approach to highlights extraction from a sport video. In *IEEE International Conference on Image Processing (ICIP 03)*, volume 1, pages 1–4, Barcelona, Spain, September 2003.
- [61] D. Harbord and S. Szymanski. Football trials. *European Competition Law Review*, 25:117–121, February 2004. URL: <http://www.market-analysis.co.uk/footballtrials.pdf>.
- [62] A. Hauptmann and R. Jin. Learning to identify video shots with people based on face detection. In *IEEE International Conference on Multimedia and Expo (ICME'03)*, volume 2, pages 293–296, Baltimore, Maryland, USA, July 2003.
- [63] A. Hauptmann, T.D. Ng, R. Baron, W. Lin, M. Chen, M. Derthick, M. Christel, R. Jin, and R. Yan. Video classification and retrieval with the Informedia digital video library system. In *Text Retrieval Conference (TREC'02)*, Gaithersburg, Maryland, USA, November 2002.
- [64] M. Höynck, T. Auweiler, and J. R. Ohm. Application of MPEG-7 descriptors for content-based indexing of sports videos. In *Visual Communications and Image Processing (VCIP 2003)*, volume 5150, pages 1317–1328, University of Italian Switzerland (USI), Lugano, Switzerland, July 2003.
- [65] J. Hu, S. G. Lim, and M. K. Brown. HMM based writer independent on-line handwritten character and word recognition. In *Sixth International Workshop on Frontiers in Handwriting Recognition*, pages 143–155, Taejeon City, Korea, August 1998.
- [66] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih. Image indexing using color correlograms. In *IEEE International Conference on Computer Vision and Pattern Recognition (ICPR'97)*, pages 762–768, Osaka, Japan, June 1997.
- [67] T. S. Huang, S. Mehrotra, and K. Ramchandran. Multimedia analysis and retrieval system (MARS) project. In *33rd Annual Clinic Library Application Data Processing – Digital Image Access Retrieval*, Urbana-Champaign, Illinois, USA, March 1996.
- [68] I. Ide, K. Yamamoto, and H. Tanaka. Automatic video indexing based on shot classification. In *Advanced Multimedia Content Processing: First International Conference (AMCP 98)*, Osaka, Japan, January 1999.

- [69] N. Inamoto and H. Saito. Arbitrary viewpoint observation for soccer match video. In *1st European Conference on Visual Media Production (CVMP'04)*, pages 21–30, London, UK, March 2004.
- [70] M. Isard and A. Blake. CONDENSATION – conditional density propagation for visual tracking. *International Journal on Computer Vision*, 29(1):5–28, 1998.
- [71] A.K. Jain. *Fundamentals of digital image processing*. Prentice Hall, 1989.
- [72] I. Jermyn, C. W. Shaffrey, and N. Kingsbury. Evaluation methodologies for Information Retrieval systems. In *Advanced Concepts for Intelligent Vision Systems (ACIVS 2002)*, Ghent, Belgium, September 2002.
- [73] T. Kawashima, K. Tateyama, T. Iijima, and Y. Aoki. Indexing of baseball telecast for content-based video retrieval. In *IEEE International Conference on Image Processing (ICIP 98)*, volume 1, pages 871–875, Chicago, Illinois, USA, October 1998.
- [74] E. Kijak, G. Gravier, P. Gros, L. Oisel, and F. Bimbot. HMM based structuring of tennis videos using visual and audio cues. In *IEEE International Conference on Multimedia & Expo (ICME03)*, volume 3, pages 309–312, Baltimore, Maryland, USA, July 2003.
- [75] E. Kijak, G. Gravier, L. Oisel, and P. Gros. Audiovisual integration for sport broadcast structuring. *Multimedia Tools and Applications*, 2004. To appear.
- [76] E. Kijak, P. Gros, and L. Oisel. Temporal structure analysis of broadcast tennis video using Hidden Markov Models. In *SPIE Storage and Retrieval for Media Databases*, pages 289–299, Santa Clara, California, January 2003.
- [77] E. Kijak, L. Oisel, and P. Gros. Hierarchical structure analysis of sport videos using HMMs. In *IEEE International Conference on Image Processing (ICIP03)*, volume 3, pages 1025–1028, Barcelona, Spain, September 2003.
- [78] A. King. A survey of methods for face detection. Technical Report 992 550 627, University of Toronto, 2003.
- [79] J. Kittler, K. Messer, W. J. Christmas, B. Levienaise-Obadia, and D. Koubaroulis. Generation of semantic cues for sports video annotation. In *IEEE International Conference on Image Processing (ICIP'01)*, volume 3, pages 26–29, Thessaloniki, Greece, September 2001.
- [80] J. G. Ko, K. N. Kim, and R. S. Ramakrishna. Facial feature tracking for eye-head controlled human computer interface. In *IEEE Analog and Digital Techniques for Electrical Engineering 1999 (IEEE TENCON 1999)*, pages 72–75, Cheju, Korea, September 1999.

- [81] A. Kokaram. *Motion Picture Restoration*. Springer, 1998.
- [82] A. C. 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 (MMSP 2001)*, pages 251–256, Cannes, France, October 2001.
- [83] D. Koller, K. Daniilidis, and H-H. Nagel. Model-based object tracking in monocular image sequences of road traffic scenes. *International Journal of Computer Vision*, 10(3):257–281, 1993.
- [84] I. Kolonias, W. Christmas, and J. Kittler. Tracking the evolution of a tennis match using Hidden Markov Models. In *International Workshop on Structural and Syntactic Pattern Recognition (SSPR 04)*, Lisbon, Portugal, August 2004.
- [85] Takashi Matsumoto Laboratory. URL: <http://www.matsumoto.elec.waseda.ac.jp/moji/moji.pdf>.
- [86] S. Lawrence, K. Bollacker, and C. L. Giles. Indexing and retrieval of scientific literature. In *Eighth International Conference on Information and Knowledge Management (CIKM 99)*, pages 139–146, Kansas City, Missouri, USA, November 1999.
- [87] C. W. Lee, H. J. Lee, S. H. Yoon, and J. H. Kim. Gesture recognition in video image with combination of partial and global information. In *Visual Communications and Image Processing (VCIP 2003)*, volume 5150, pages 458–466, University of Italian Switzerland (USI), Lugano, Switzerland, July 2003.
- [88] H. Lee and A. F. Smeaton. Designing the user interface for the Físchlár digital video library. *Journal of Digital information : Digital libraries, Hypermedia systems, Usability of digital information*, 2(4), 2002. URL: <http://jodi.ecs.soton.ac.uk/Articles/v02/i04/Lee/>.
- [89] J. J. Lee, J. Kim, and J. H. Kim. Data-driven design of HMM topology for on-line handwriting recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(1):107–121, 2001.
- [90] M. W. Lee, I. Cohen, and S. K. Jung. Particle filter with analytical inference for human body tracking. In *IEEE Workshop on Motion and Video Computing*, pages 159–165, Orlando, Florida, USA, December 2002.
- [91] R. Leonardi, P. Migliorat, and M. Prandini. Semantic indexing of sports program sequences by audio-visual analysis. In *IEEE International Conference on Image Processing (ICIP 03)*, volume 1, pages 9–12, Barcelona, Spain, September 2003.
- [92] M.S. Lew, N. Sebe, and P. Gardner. *Principles of Visual Information Retrieval*, pages 163–196. Springer, 2001.

- [93] B. Li and I. Sezan. Semantic sports video analysis : approaches and new applications. In *IEEE International Conference on Image Processing (ICIP03)*, volume 1, pages 17–20, Barcelona, Spain, September 2003.
- [94] T. Liu, H.-J. Zhang, and F. Qi. A novel video keyframe extraction algorithm. In *IEEE International Symposium on Circuits and Systems (ISCAS 2002)*, volume 4, pages 149–152, Scottsdale, Arizona, USA, May 2002.
- [95] P. J. Macer and P. J. Thomas. Video storyboards: summarising video sequences for indexing and searching of video databases. In *IEE Colloquium on Intelligent Image Databases*, volume 2, pages 1–5, London, UK, May 1996.
- [96] D. J. C. MacKay. *Introduction to Monte Carlo Methods*, pages 175–204. MIT Press, 1999.
- [97] S. Marlow, D. Sadlier, N. O'Connor, and N. Murphy. Audio processing for automatic TV sports program highlights detection. In *Irish Signals and Systems Conference (ISSC 02)*, Cork, Ireland, June 2002.
- [98] Mathworld. URL: <http://mathworld.wolfram.com/Point-LineDistance2-Dimensional.html>.
- [99] S. Mavromatis, J. Baratgin, and J. Sequeira. Reconstruction and simulation of soccer sequences. In *Model-based Imaging, Rendering, image Analysis and Graphical special Effects (MIRAGE 2003)*, Rocquencourt, France, March 2003.
- [100] D. Metzeler and R. Manmatha. An inference network approach to image retrieval. In *3rd International Conference on Image and Video Retrieval (CIVR 04)*, volume 3115, pages 43–50, Dublin, Ireland, July 2004.
- [101] A. Moore. URL: <http://www-2.cs.cmu.edu/~awm/tutorials/kmeans.html>.
- [102] A. Moore. URL: <http://www-2.cs.cmu.edu/~awm/tutorials/gmm.html>.
- [103] C. N. Moores. Datacoding applied to mechanical organization of knowledge. *American Documentation*, 2:20–32, 1951.
- [104] A. Nefian. *A Hidden Markov model-based approach for face detection and recognition*. PhD thesis, Georgia Institute of Technology, 1999.
- [105] NetByTel. URL: <http://www.netbytel.com/literature/e-gram/technical3.htm>.
- [106] C.-W. Ngo, T.-C. Pong, and H.-J. Zhang. Recent advances in content-based video analysis. *International Journal of Image and Graphics*, 1(3):445–468, 2001.

- [107] K. Nummiaro, E. Koller-Meier, and L. Van Gool. A color-based particle filter. In *First International Workshop on Generative-Model-Based Vision, in conjunction with ECCV02*, pages 53–60, Copenhagen, Denmark, June 2002.
- [108] K. Nummiaro, E. Koller-Meier, and L. Van Gool. Object tracking with an adaptive color-based particle filter. In *Symposium for Pattern Recognition of the DAGM*, pages 353–360, Zürich, Switzerland, September 2002.
- [109] J. H. Oh, K. A. Hua, and N. Liang. A content-based scene change detection and classification technique using background tracking. In *Proceedings of SPIE: Storage and Retrieval for Media Databases*, volume 3969, pages 254–265, San Jose, California, USA, January 2000.
- [110] D. E. O’Leary. Semantic ambiguity in expert systems: The case of deterministic systems. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 92)*, volume 682, pages 304–313, Berlin, Heidelberg, July 1992.
- [111] Informal Ministerial Conference on Broadcasting. The impact of transfrontier broadcasting services on television markets in individual member states, February 2004. URL: http://www.obs.coe.int/online_publication/transfrontier_tv.pdf.en.
- [112] H. Pan, P. Van Beek, and M. Sezan. Detection of slow-motion replay segments in sports video for highlights generation. In *IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP 01)*, volume 3, pages 1649–1652, Salt Lake City, Utah, USA, May 2001.
- [113] R. Pass, G. Zabih, and J. Miller. Comparing images using color coherence vectors. In *4th ACM Conference on Multimedia*, pages 65–73, Boston, Massachusetts, USA, November 1996.
- [114] K. A. Peker, R. Cabasson, and A. Divakaran. Rapid generation of sports video highlights using the MPEG-7 motion activity descriptor. In *Symposium of Electronic Imaging: Science and Technology: Storage and Retrieval for Media Databases*, pages 318–323, San Jose, California, USA, January 2002.
- [115] A. Pentland, R. W. Picard, and S. Sclaroff. Photobook: tools for content based manipulation of image databases. In *The International Society for Optical Engineering. (Storage and Retrieval for Image and Video Databases II)*, volume 2185, pages 34–47, San Jose, California, USA, February 1994.
- [116] P. Perez, A. Blake, and M. Gangnet. Jetstream: Probabilistic contour extraction with particles. In *International Conference on Computer Vision (ICCV 2001)*, pages 524–531, Vancouver, Canada, July 2001.

- [117] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Colour based probabilistic tracking. In *European Conference on Computer Vision 2002 (ECCV 2002)*, pages 661–675, Copenhagen, Denmark, May 2002.
- [118] P. Perez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proceedings of the IEEE (Special issue on State Estimation)*, 92(3):495–513, March 2004.
- [119] M. Petkovic and W. Jonker. Content-based video retrieval by integrating spatio-temporal and stochastic recognition of events. In *IEEE Workshop on Detection and Recognition of Events in Video*, pages 75–82, Vancouver, Canada, July 2001.
- [120] S. Pfeiffer, R. Lienhart, S. Fischer, and W. Effelsberg. Abstracting digital movies automatically. *Journal of Visual Communication and Image Representation*, 7(4):345–353, 1996.
- [121] M. J. Pickering and S.M. Riiger. Multi-timescale video shot change detection. In *Tenth Text Retrieval Conference*, pages 275–280, Gaithersburg, Maryland, USA, November 2001.
- [122] C. A. Poynton. *A technical introduction to digital video*. Wiley, 1996.
- [123] Y. Qi, T. Liu, and A. Hauptmann. Supervised classification of video shot segmentation. In *IEEE Conference on Multimedia and Expo (ICME'03)*, volume 2, pages 689–692, Baltimore, Maryland, USA, July 2003.
- [124] L. R. Rabiner and B. H. Juang. An introduction to Hidden Markov Models. *IEEE Acoustics, Speech & Signal Processing Magazine*, 3(1):4–16, 1986.
- [125] 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 (ICASSP 2004)*, volume 3, pages 621–624, Montreal, Canada, May 2004.
- [126] 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)*, volume 3115, pages 88–97, Dublin, Ireland, July 2004.
- [127] P. Rea. Automated migration calculation. Master's thesis, University of Oxford, Pembroke College, St. Aldates, Oxford, 2003.
- [128] S. K. Riis. *Hidden Markov Models and Neural Networks for Speech Recognition*. PhD thesis, Department of Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, DK-2800, 1998.

- [129] M. Roach, J. Mason, L-Q. Xu, and F. W. M. Stentiford. Recent trends in video analysis: a taxonomy of video classification problems. In *6th IASTED International Conference on Internet and Multimedia Systems and Applications*, Kauai, Hawaii, USA, August 2002.
- [130] S. Robertson. *Lecture Notes in Computer Science: Lectures on Information Retrieval*, volume 1980/2001, chapter 4, pages 81–92. SpringerVerlag, September 11–15 2000.
- [131] E. Salari and Z. Ling. Texture segmentation using hierarchical wavelet decomposition. *Pattern Recognition*, 28:1819–1824, 1995.
- [132] F. Schaffalitzky and A. Zisserman. Automated scene matching in movies. In *Challenge of Image and Video Retrieval (CIVR'02)*, pages 186–197, London, UK, July 2002.
- [133] Scope. URL: http://www.science.ie/scopetv/home/index.asp?section_id=588.
- [134] N. Sebe and M.S. Lew. Color based retrieval. *Pattern Recognition Letters*, 22:223–230, February 2001.
- [135] C. W. Shaffrey. *Multiscale Techniques for Image Segmentation, Classification and Retrieval*. PhD thesis, St Edmunds College, University of Cambridge, September 2003.
- [136] J. R. Smith and S-F. Chang. Visualeek: A fully automated content-based image query system. In *Fourth ACM international conference on Multimedia*, pages 87–98, Boston, Massachusetts, USA, November 1996.
- [137] M. Smith and T. Kanade. Video skimming and characterisation through the combination of image and language understanding techniques. In *IEEE Computer Vision and Pattern Recognition (CVPR 97)*, pages 775–781, San Juan, Puerto Rico, June 1997.
- [138] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis and Machine Vision*. Chapman & Hall, 1993.
- [139] H. Stark and J. W. Woods. *Probability and Random Processes with Applications to Signal Processing*. Prentice Hall, third edition, 2002.
- [140] T. Starner and A. Pentland. Real-time ASL recognition from video using HMMs. In *International Symposium on Computer Vision*, pages 21–23, Coral Gables, Florida, USA, November 1995.
- [141] G. Sudhir, J. C. M. Lee, and A. K. Jain. Automatic classification of tennis video for high-level content-based retrieval. In *IEEE International Workshop on Content-Based Access of Image and Video Database*, pages 81–90, Bombay, India, January 1998.

- [142] K. Takahashi, H. Yasuda, and T. Matsumoto. A fast HMM algorithm for on-line handwritten character recognition. In *Fourth International Conference on Document Analysis and Recognition (ICDAR '97)*, volume 1, pages 369–375, Ulm, Germany, August 1997.
- [143] T. Taki, J. Hasegawa, and T. Fukumura. Development of motion analysis system for quantitative evaluation of teamwork in soccer games. In *IEEE International Conference on Image Processing (ICIP 96)*, pages 815–818, Lausanne, Switzerland, September 1996.
- [144] Y-P. Tan, D. D. Saur, and S. R. Kulkarni. Rapid estimation of camera motion from compressed video with applications to video annotation. *IEEE Transactions on Circuits and Systems for Video Technologies*, 10:133–146, 2000.
- [145] D. O. Tanguay. Hidden Markov Models for gesture recognition. Master's thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, August 1995.
- [146] C. Thies, V. Metzler, and T. Aach. Content-based image analysis: Object extraction by data-mining on hierarchically decomposed medical images. In *SPIE Medical Imaging*, volume 5032, San Diego, California, USA, February 2003.
- [147] P. Toft. *The Radon Transform - Theory and Implementation*. PhD thesis, Department of Mathematical Modeling, Section for Digital Signal Processing, Technical University of Denmark, Lyngby, Denmark, 1996.
- [148] V. Tovinkere and R. J. Qian. Detecting semantic events in soccer games: towards a complete solution. In *IEEE International Conference on Multimedia and Expo (ICME 01)*, pages 833–836, Tokyo, Japan, August 2001.
- [149] T. Tuytelaars, L. Van Gool, M. Proesmans, and T. Moons. The cascaded Hough transform as an aid in aerial image interpretation. In *Proceedings of the Sixth International Conference on Computer Vision*, pages 67–72, Bombay, India, January 1998.
- [150] S. Uchihashi, J. Foote, A. Girgensohn, and J. Boreczky. Video Manga: Generating semantically meaningful video summaries. In *ACM Multimedia*, pages 383–392, Orlando, Florida, USA, October 1999.
- [151] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, second edition, 1979.
- [152] C. J. van Rijsbergen. *Lecture Notes in Computer Science: Lectures on Information Retrieval*, volume 1980/2001, chapter 1, pages 1–20. SpringerVerlag, September 11–15 2000.

- [153] N. Vasconcelos and A. Lippman. A Bayesian video modeling framework for shot segmentation and content characterization. In *IEEE Workshop on Content-based Access to Image and Video Libraries (CVPR 97)*, pages 59–64, San Juan, Puerto Rico, June 1997.
- [154] J. Vermaak, M. Gangnet, A. Blake, and P. Perez. Sequential Monte Carlo fusion of sound and vision for speaker tracking. In *IEEE International Conference on Computer Vision (ICCV 2001)*, volume 1, pages 741–746, Vancouver, Canada, July 2001.
- [155] J. Y. A Wang and E. H. Adelson. Representing moving images with layers. *The IEEE Transactions on Image Processing Special Issue: Image Sequence Compression*, 3(5):625–638, September 1994.
- [156] G. Welch and G. Bishop. An introduction to the Kalman filter. Technical Report TR95-041, Department of Computer Science, University of North Carolina, 2004.
- [157] C. Wu, Y. F. Ma, H. J. Zhang, and Y. Z. Zhong. Events recognition by semantic inference for video. In *International Conference on Multimedia and Expo (ICME 2002)*, volume 1, pages 805–808, Swiss Federal Institute of Technology, EPFL, Lausanne, Switzerland, August 2002.
- [158] M. Wu, W. Wolf, and B. Liu. An algorithm for wipe detection. In *IEEE International Conference on Image Processing (ICIP 98)*, volume 1, pages 893–897, Chicago, Illinois, USA, October 1998.
- [159] H. Li X. Gibert and D. Doermann. Sports video classification using HMMs. In *IEEE International Conference on Multimedia and Expo (ICME '03)*, volume 2, pages 345–348, Baltimore, Maryland, USA, July 2003.
- [160] L. Xie, S-F. Chang, A. Divakaran, and H. Sun. Structure analysis of soccer video with Hidden Markov Models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'02)*, volume 4, pages 4096–4099, Orlando, Florida, USA, May 2002.
- [161] G. Xu, Y.-F. Ma, H.-J. Zhang, and S. Yang. Motion based event recognition using HMM. In *IEEE International Conference on Pattern Recognition (ICPR 02)*, volume 2, pages 831–834, Quebec City, QC, Canada, August 2002.
- [162] G. Xu, Y-F. Ma, H-J. Zhang, and S. Yang. A Hidden Markov Model based semantic analysis framework for sports game event detection. In *IEEE International Conference on Image Processing (ICIP 03)*, volume 1, pages 25–28, Barcelona, Spain, October 2003.

- [163] J. Yamoto, J. Ohya, and K. Ishii. Recognising human action in time-sequential images using Hidden Markov Models. In *Proceedings of IEEE Computer Vision and Pattern Recognition*, pages 379–385, Champaign, Illinois, USA, June 1992.
- [164] H. Yasuda, K. Takahashi, and T. Matsumoto. A discrete HMM for online handwriting recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 14(5):675–689, 2000.
- [165] S. Yeterian. Content aware movement keyframes for sport. Master’s thesis, Eurecom Institute and Swiss Institute of Technologies, 2004.
- [166] D. Yow, B. Yeo, M. Yeung, and G. Liu. Analysis and presentation of soccer highlights from digital video. In *Proceedings of Asian Conference on Computer Vision (ACCV 95)*, pages 499–503, Singapore, December 1995.
- [167] R. Zabih, J. Miller, and K. Mai. Feature based algorithms for detecting and classifying scene breaks. In *4th ACM International Conference on Multimedia*, pages 189–200, Boston, Massachusetts, USA, November 1995.
- [168] R. Zabih, J. Miller, and K. Mai. A feature-based algorithm for detecting and classifying production effects. *ACM Journal of Multimedia Systems*, 7(2):119–128, 1999.
- [169] H. J. Zhang, A. Kankanhalli, and S. W. Smoliar. Automatic partitioning of full-motion video. *ACM Multimedia System*, 1:10–28, 1993.
- [170] H. J. Zhang, C. Y. Low, Y. Gong, and S. Smoliar. Video parsing using compressed data. *SPIE Image and Video Processing II*, 2182:142–149, February 1994.
- [171] H. J. Zhang, J. H. Wu, D. Zhong, and S. W. Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern Recognition*, 30(4):643–658, 1997.
- [172] D. Zhong and S-F. Chang. Structure analysis of sports video using domain models. In *IEEE International Conference on Multimedia and Expo (ICME '01)*, pages 182–186, Tokyo, Japan, August 2001.
- [173] H. Zhong, M. Visontai, J. Shi, H. Zhong, J. Shi, and M. Visontai. Detecting unusual activity in video. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'04)*, volume 2, pages 819–826, Washington, District of Columbia, USA, June 2004.
- [174] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *IEEE International Conference on Image Processing (ICIP 98)*, volume 1, pages 866–870, Chicago, Illinois, USA, October 1998.

-
- [175] Z. Zivkovic, F. van der Heijden, M. Petkovic, and W. Jonker. Image segmentation and feature extraction for recognizing strokes in tennis game videos. In *Seventh Annual Conference of the Advanced School for Computing and Imaging (ASCI 2001)*, pages 262–267, Heijen, The Netherlands, May 2001.