Object Tracking In 3D

Progress Report

Adam Hartshorne

December 2004

Abstract

A system will be investigated and implemented that, with the use of two video cameras and software, observe the motion of an object and form a model of its movement. This model of the motion will be animated in a 3D Virtual World.

Contents

List of Figures

Chapter 1

Introduction

This report outlines the progress that has currently been made in attempting to produce a solution to the problem summarized in the abstract, and describes the challenges which are still required to be overcome to complete the project.

In addition, this report summarizes the changes to the original specification that are have been employed, due to shortcomings that have been identified or deemed necessary in order to solved the overall problem. The amended specification, including a revised timetable for the work that still is required, is included in Appendix A.

This report exceeds the 2000 word guideline, however this has been necessary in order to outline the extensive research and implementation that has been undertaken to date. This fact has been revealed to the project supervisor, and deemed acceptable in principle.

1.1 Overview

Although the abstract purposes a general image-tracking problem, as outlined in the specification document the project being undertaken is now specifically aimed at tracking cricket balls during a segment of a delivery.

The system is envisaged to work by placing two analogue cameras around a cricket pitch. The exact placement of the cameras is to be part of the project. The cameras will record a delivery, and the system should be able to track the ball throughout its motion in the field of vision of the two cameras.

1.2 Background

During initial research for this project, it has been found that a great deal of research is currently being conducted into object tracking systems. There are also organisations, such as Sportvision, whose products range are dedicated to enhancing the spectating and participation experience of a variety of different sports.

Hawkeye is at present the most prominent system that exists in this field. As set out in the specification document, the motivation for this project is partly due to this system. The system has the ability to track a moving object to an accuracy of approximately 1mm, and also predict the future path of this object. This is used to give the television viewer a better indication if LBW decisions made during the course of a cricket match are correct.

This system uses a set of 6 high-speed cameras and requires a team of people to set up and operate it. For these reasons, the system is extremely costly, and is therefore only presently in operation in certain high profile matches.

However, Hawkeye is not the only commercial ball tracking sports system. In American, the sports television network ESPN have employed a system entitled K-Zone, based upon Gueziec′s [1] paper. It is established on similar principles as that of Hawkeye, to determine if a particular pitch is a strike during a game of baseball. The system uses three PCs connected to three video cameras, two observing the flight of the baseball, and the third one observing the batter for correct sizing of the strike zone.

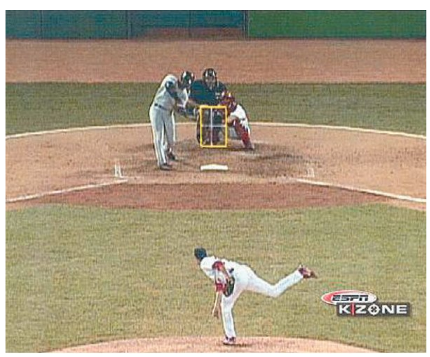

Figure 1.1: K-Zone in use during a televised game.

Both of these systems require multiple cameras, PCs and operators to track the motion of the thrown ball. However, only the Hawkeye system actually predicts the future motion of the ball.

The technical details that have been ascertained about these products have provided invaluable foundational knowledge. However, the conceived project solution does not purely implement the various algorithms underpinning these systems for a variety of reasons.

Chapter 2

Technical Challenges

2.1 Simplifications

At the time of the initial filming, it was deemed to be unrealistic to create a complete system with high accuracy calibration and that was robust to a moving batsman. This was due to a limited knowledge of image processing and more specifically object tracking. In addition, certain hardware restrictions exist which can′t be overcome. Therefore the system being developed, has the following simplifications:-

1. The system does not act on streaming video - Hardware restrictions do not allow multiple cameras to be easily attached to a single PC. This is especially prevalent with the use of analog cameras, where two capture cards would be required. [1]

2. Object tracking does not take place in real time - Developing a real time tracking system would add another level of complexity, which has been deemed inappropriate for this project. It should also be noted that Hawk-eye and K-Zone do not currently operate in real time.

3. The system is developed with no batsman in mind - The cameras are focused on an area of the pitch where for the vast majority of deliveries a batsman will not be in the shot.

4. The cameras are regarded as perfectly synchronised - This is not currently a legitimate assumption, as additional specific hardware is required to achieve this synchronisation. Captured video may be edited to attempt to achieve some degree of synchronisation.

During the research that has been undertaken for this project, ideas of how to develop a system which doesn′t contain these simplifications have been formulated. It is intended that the final report will contain a description of the

[1] 1 and 2 are interrelated

techniques that have been considered, and that would overcome the limitations of the current proposed system.

2.2 Camera Positioning & Filming Issues

The abstract does not clearly state that the motion capturing process is to be part of the project. However, it has been found to be an essential part of it, because is it not possible to gain access to the suitable footage. The scope of this particular task alone could result in a comprehensive investigation encompassing a variety of aspects, such as the effect of the camera position on calibration error.

However, in order to be able to progress with the development of a solution to the overall problem, an arrangement that produced satisfactory results was required. Various configurations were examined, with the following factors [2] being considered for each:

1. Practicality of camera positioning.
2. Allowing a delivery to be made in an unimpeded manner.
3. Sufficiently large captured image of the ball enabling successful tracking to be undertaken. [3]
4. Sufficiently clear captured image of the ball. [4]
5. Sufficient number of captured frames which contained the ball in motion. [5]

A configuration with a 60^o separation angle between the cameras was found to produce the optimum balanced performance, when assessed against the various criteria. Each camera is located on a path at 30^o to an imaginary line from a position fixed at 4.5m in front of middle stump (batsmans end) to the middle stump (non-striker′s end). The cameras were required to be angled 10^o downwards in order to view the pitch using this configuration.

It is intended that the system being developed will not just be limited to this particular camera configuration. In spite of this, this arrangement was used for all the initial video capture.

[2] The description of the factors are noticeably vague. This is due to actual required numerical values for the various parameters been unknown at the present time. The figures were produced from a judgement of required minimums of the different parameters.

[3] The required ball representation is a minimum of 15 pixels.

[4] The required number of frames with a blurred image of the ball in motion needs to be less than 10% of the total.

[5] A minimum of 5 frames at 30fps.

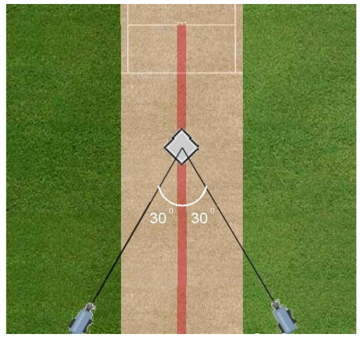

Figure 2.1: Overhead representation of camera configuration.
(NB:- The position of the cameras in reality were much further from the pitch)

2.3 Summary of Component Design & Related Issues

The entire process, from delivery of the ball to visualising it′s trajectory, has been broken down into six stages, which are allied in the following way:-

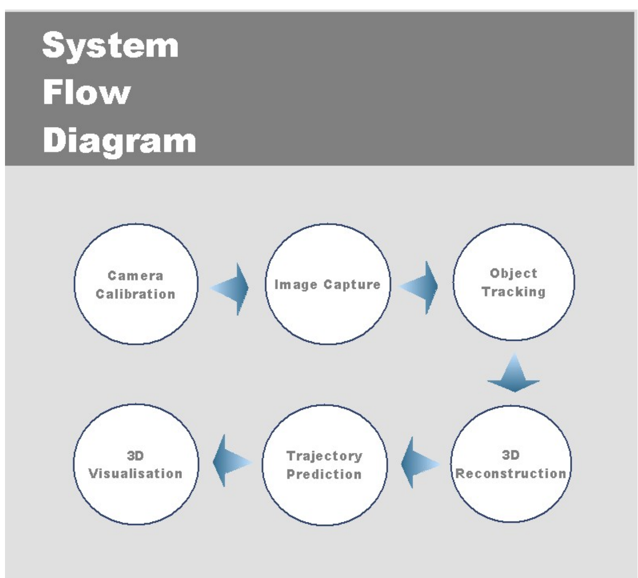

Figure 2.2: Overview of system components.

During the past three months various techniques, strategies and algorithms related to each of the identified stages have been researched. Prototypes for the majority of ideas considered in depth have been developed and implemented with varying degrees of success. Nonetheless, a single satisfactory solution for each stage has been established and examined with the use of some of the captured video clips.

In the following sections these techniques, and issues related with their use, are briefly discussed. However, specific in depth technical information has been omitted at this stage. The final report will provide a detailed description of techniques, the motivation behind their use, and the mathematical derivations required, where appropriate, to produce the formulae implemented in the final application.

Although good design practices are being employed, after discussion with my project supervisor, UML and other formal design methodologies are not been utilized. As a consequence of the nature of the project, where many aspects require a trail and error approach to form a solution, it has been deemed that these methodologies are not suitable.

2.3.1 Camera Calibration

In order to map points from the 2D images to real 3D space, some kind of information regarding the camera is required. The problem is creating a camera for the two views that is robust against factors such as the focal length, lens distortion and camera positions. It was decided that it should also be convenient to calculate in the sense that it requires a minimal number of external factors, such as camera angles, to be known. For practical reasons, the camera calibration should not be needed to be done every delivery, but be performed once for a whole series of deliveries.

Two different mathematical based approaches, which fulfil these criteria, have been investigated and found to produce satisfactory results. They both require the use of a calibration object.

Figure 2.3: Calibration object used in current filming.
(Can be used to select upto 45 different calibration points)

In [2], Crowley, Bobet & Schmid outline a complete calibration algorithm that directly obtains the perspective transformation matrix for a single camera in isolation. It uses six reference points, where four define a reference plane and a further two are chosen to be a linear combination of the four. The method produces the transformation matrix for points relative to the coordinate system described by the calibration object.

A similar technique to the [2], by Sid-Ahmed & Boraie [3], uses a more abstract mathematical approach. The advantage of this approach is that although the algorithm only requires six points to gain a calibration reading, it is readily extended to many more points. Thus increasing the accuracy of the calibration of the system. An over-determined system, such as one that may be created by this method, is easily solved using a least-squares approach.

An additional transform from world to box coordinates is still required when using either technique, to include the box dimensions, location and orientation. Following the calibration and addition transformations on coordinates from either camera, coordinates are produced that have a consistent set of global axes.

2.3.2 Image Capture

Once the camera configuration is decided and calibration images have been recorded, the capture of the required footage is a straightforward and trivial issue of camera operation.

However, some post capture modification to the footage has been found to be required. If we assume a maximum delivery speed of 90mph (which a first class cricketers can achieve), and consider the fact that video cameras available in the high street operate at 30fps, it is only possible to capture a small number of frames, even if the whole of the delivery could be captured.

Research has uncovered algorithms to alter the frame rates of some recorded video footage. They use the fact that a frame is in reality two half images, called fields. These fields can be separated out and converted into frames of their own. This results in footage, which has double the frame rate, but half the resolution of the original. Although, one of these algorithms could have been implemented directly, equivalent functionality has been found to be available in VirtualDub, a video processing package.

Furthermore, this package has also been utilized to split the footage up into individual deliveries. In addition,this division has also been undertaken in such a way as to attempt to make the corresponding frames from the two views of a particular delivery relate to the same moment in time. This is an attempt at synchronising the two views, although it is understood that it is not possible to do this to any real accuracy.

2.3.3 Object Tracking

The challenge is in reality to find the centroid of the ball in each frame. The difficulty of this may be increased by:-

1. Movement in the image by objects other than the ball.
2. Blurring of the tracked object in the image.
3. The fact that a sphere is displayed as an ellipse when viewed in 2D.

Various tracking strategies were initially researched to different extents. The particular strengths and weaknesses of each, with regards to the project requirements, were fully considered. A blob tracking approach was deemed to be the most suitable for this particular problem. A six-stage algorithm based upon this concept has been developed to track the ball.[6]

Currently it is not appropriate to describe these stages in extensive depth, however a few remarks about each are presented in this document. This is to enable the reader to gain an appreciation of the objective each stage attempts to fulfil.

1. The current and previous captured frames are split into their RGB components. The splitting of constituent colours has been found to be essential to the process. Without it, the following steps are inefficient at removing noise. From these three components only the green element is retained. [7]
2. The green components of each frame are compared, using an XOR function, with the result being the difference between these two frames. A simple subtraction between each frame was investigated, but was found to produce inferior results
3. A median filter is applied to the remaining image, in order to remove the majority of the image noise that exists at this stage.
4. A threshold technique is employed to remove remaining noise pixels. The particular threshold procedure employed acts upon a range of values, rather than a single value. An algorithm calculates the min and max values of this range automatically, based upon a histogram distribution of the pixel values.
5. A blob-labelling approach joins remaining connected pixels together in blobs. Blobs that are not within a set range [8] are discarded.
6. The position of the approximate centroid of a blob is calculated by finding the average value of the x and y coordinates of pixels in that blob.

[6] The functionality for the first four stages been developed from base methods that have been found to be available through the Java JAI (Java Advanced Imaging) API.

[7] Processing all three components seperately and combining them afterwards produced an intolerablely slow process. It was found through experimentation that the following stages produced the best results when using the green component.

[8] This a result of the conversion of the real size of the ball to an expected represented size using calibration information.

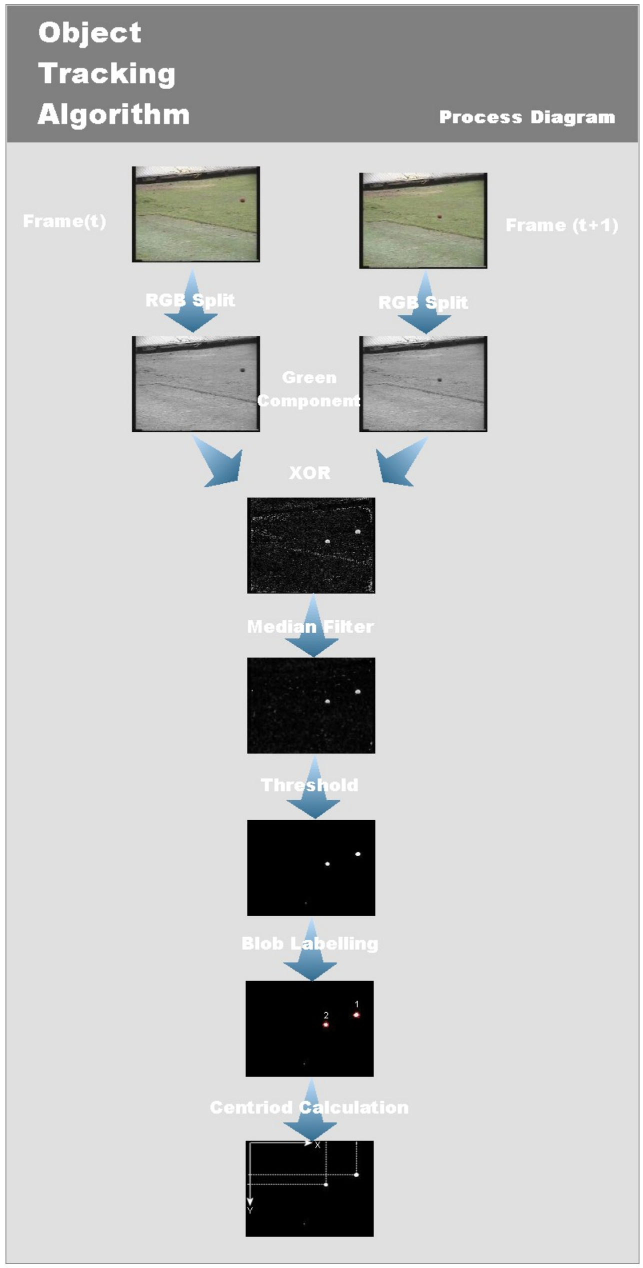

Figure 2.4: Overview of the ball tracking algorithm.
(containing typical output from each stage)

2.3.4 3D Position Reconstruction

Once we have the 2D co-ordinates of the ball in terms of their pixel values in each frame, it is necessary to combine the values from the camera views to form an estimate of where the ball is located in 3D space.

This can be done geometrically by finding the intersection of two lines in space. A line is formed by considering a projection from a camera′s optical origin through the image plane, taking into account the calculated ball position (in 2D) from the image that the camera has recorded. However, quantisation (due to the camera calibration) and ball location errors have been found to result in the lines not intersecting perfectly.

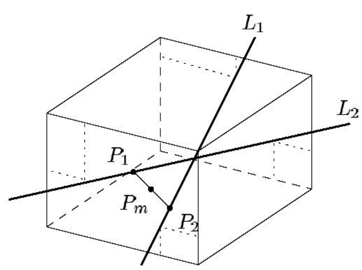

Figure 2.5: Lines of sight L_1,L_2 with closest points P_1,P_2 & midpoint P_m shown.

In order to overcome this problem, a slightly more complex solution has been required. It has be formed based upon two simple geometrical principles:-

1. A set of points equidistant from a line (with distance R) is a cylinder.
2. The shortest line between two non-intersecting lines will be perpendicular to both.

From these two facts, a mathematical representation of the shortest possible line that can join the two lines can be found. The mid-point of this line is calculated, and it is stated that this is the 3D position of the ball.

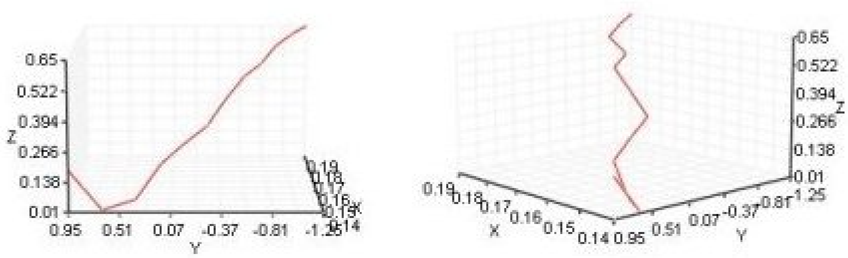

Figure 2.6: Result of utilisation of the 3D Position Reconstruction algorithm. (shown from two different views)

2.3.5 Motion Prediction

Given the 3D points, which will contain uncertainty introduced by processing errors, it is necessary to manipulate these points to fit a smooth trajectory. In addition, an estimate of the position of the ball outside the area of the tracking is required.

Several purely mathematical curve-fitting techniques have been considered, including least square reduction of B-Splines [4] and least square curve fitting with confidences [5]. These techniques successfully fitted a 3D curve through the measured points. However, they was found to perform poorly as a prediction mechanism.

As an alternative, a Kalman filter has been designed to fit the tracked motion. It can also be used to predict states outside the range of those tracked. After consultation with [6] publication, a physics based model for the motion of the ball in the vertical plane has been formulated. The formulae presented in [6] for this motion are strongly coupled with one another. A Runge Kutta numerical integration technique was required to produce uncoupled equations that fitted the form required by the Kalman Filter base equations.

[6] outlines how to calculate the extent of any the lateral movement. However, this requires a measurement of the rotational speed of the ball. With the hardware available and the type of object-tracking algorithm that has been used this is not a viable option. Therefore, the lateral movement component of the model is based solely on variables calculated through trial and error.

2.3.6 3D Visualisation

Using basic functionality available in the Java 3D API, the animation of 3D points has been found to be a relatively straightforward procedure. At present the animation is extremely basic, with a simple world containing an unsophisticated representation of the pitch and the ball.

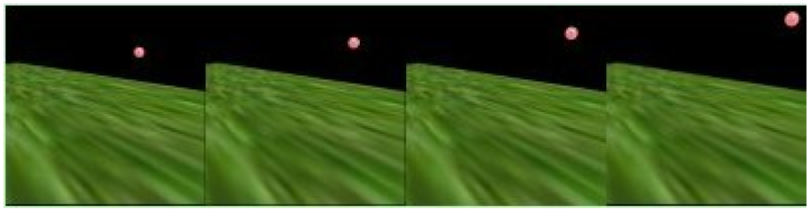

Figure 2.7: Example of a visualised delivery.

However, research has been conducted into advanced Java 3D functionality, which allows 3D models created in third party modelling packages, such as 3D Studio Max, to be utilised in a java application.

This functionality has been tested with some free unrelated models, including a car and a plane. The methods, which handle this process, appear to work faultlessly. The rendered version of a model loaded into a Java 3D world produces a representation which is indistinguishable from the model viewed using the intended the third party package. A model of cricket stadium has been found, although a small fee is required for it′s use.

Chapter 3

Current Project Situation & Impending Work Information

In relation to the timetable outlined in the original version of my specification document, I am at present in advance of my predicted progress. I intended to be at the current position in the development cycle in the early stages of next term.

A working prototype has been produced for each of the solutions to the identified sub-tasks. However, at present these individual units have not been rigorously implemented and combined to produce a final solution. The majority are unconnected, will only operate on simple inputs and contain little error handling.

3.1 Graphic User Interface

Consequently, the primary work to be undertaken next term will be to use a professional design attitude to fully implement the algorithms in order to develop a complete GUI based application.

In addition, it proposed that as part of the final GUI driven application the ability to allow deliveries to be compared will be realized. Statistics about each delivery, or group of deliveries, will be available, such as the speed or distribution of deliveries.

3.2 Enhanced Animation

The present animation that displays the tracked and predicted motion is extremely simple. As previously mentioned, a method of importing 3D models originally created in third party software has been researched and tested. It is intended that this approach will be incorporated into the final solution, in order to produce a more aesthetically pleasing and realistic animation.

Figure 3.1: Proposed 3D model for use in the visualisation of a tracked delivery.

3.3 Error Quantification

An investigation into the magnitude of the errors produced by the various units will be undertaken. It is intended that this information will be gathered by conducting addition filming. This new set of captured deliveries will use increased position indication on and around the pitch. This will allow a determination of the exact position of the ball at specific points, for instance when the ball bounces and passes the stumps.

It is essential to know the magnitude of errors of any system. However, this analysis will serve several other purposes.

1. It will allow the most accurate algorithm to be selected, in cases where several techniques have been thoroughly investigated and it has not been possible to ascertain the superior method.

2. Variables exist in the Kalman filter, which require some initial estimate of the variance of particular errors. At present, satisfactory values for these parameters have been found through a trail and error approach. The error analysis may be incorporated into the Kalman filter, by using some of the finding as the values for these error based variables.

3.4 Tracking Algorithm Developments

The possibility of using the Kalman filters prediction mechanism to reduce the search area considered as part of the object tracking algorithm will be investigated. It is anticipated this increased efficiency will result in a faster tracking method. From the prediction made by the Kalman filter of the next 3D state, it should be possible to be decomposed it into the two 2D coordinates. An area on each image, based upon the appropriate coordinate, may be searched, rather than the complete image.

If this is possible, it may also be feasible to consider all the RGB elements as part of the tracking, without increasing the running time of the algorithm to an unexpectable level. Rather than disposing of the red and blue components, it may be possible to use the tracking algorithm on all three components. Following such an operation, the three sets of results may be combined to provide a better segementation result. This hopefully would result in an increase in the accuracy of the centriod calculation.

Appendix A Specification (Last Updated 05/12/03)

A.1 Introduction

This document describes the project, the final goals, deliverables and an outline of how the project will be carried out. This document should be thought of as a contract between the student and supervisor, however the specification should be expected to change throughout the year.

A.2 Description

A.2.1 Motivation

The motivation behind this project comes from firstly a personal interest in the area of image processing from video footage. More specifically how this field of computer science can be implemented in order to enhance the spectating / participating experience and coaching of various sports.

At present a system called Hawkeye exists, which tracks a cricket ball during each delivery of a match where it is in use. The system has the ability to track the moving object to an accuracy of approximately 1mm, and also predict the future path of the object. This is used to give the television viewer a better indication if certain decisions made during the course of a game were correct.

This system uses a set of 6 high-speed cameras and requires a team of people to set up and operate the system. For these reasons the system is extremely costly, and is therefore only presently in operation in certain matches high profile matches. The system is most certainly out of reach of use by amateur players and their clubs.

A more affordable system could allow more players to experience and gain valuable information from using this type of tracking and prediction system. For example, to analyse the distribution of a particular bowlers deliveries, as use as part of a coaching program.

A.2.2 Implementation

The system is envisaged to work by placing two analogue cameras around a cricket pitch. The exact placement of the cameras will be part of the project. The cameras will record a delivery, and the system should be able to track the ball throughout its motion in the field of vision of the two cameras.

If required a prediction of where a particular delivery outside the range of the system, for example to determine if it would have hit the stumps, is to be part of the functionality available. This prediction will be performed automatically by extrapolating the computed trajectory of the ball past the last frame where the ball was deflected, using physical principles of a ball in flight.

The exact location of a ball at a particular time will be calculated by extracting the balls location in each of the two frames. Triangulation between the two cameras and the ball will then be used to locate the ball in 3D space. Some sort of calibration process, automatic or manual is required in order to calculate the exact location in real world coordinates.

An animation of the motion of the ball will be presented. The system will allow the user to view this animation from different views and at different rates.

A.3 Objectives

The primary goals of the project, in order are:

1. Extract the ball location in a noisy scene
2. Calibrate the ball locator
3. Determine ball location in 3D space from two camera views
4. Predict the balls trajectory (including bounces)
5. Display the motion of the ball as a 3D interactive animation
6. Determine the errors in the system

However, if my primary objectives are achieved within the time scale, possible extensions and improvement with be investigated. For example, to enable the system able to locate first and last frames of interest.

The following simplifications have been deemed necessary to complete the project:-

1. The system does not act on streaming video.
2. Object tracking does not take place in real time.
3. The system is developed with no batsman in mind.
4. The cameras are regarded as perfectly synchronised.

A.4 Methods

Currently the exact methods required to achieve the objectives are not known. However the aforementioned objectives appear to be independent from one another, and so can be completed separately or in parallel. However, it is felt that is would be better to achieve initial prototypes of each objective separately in the order specified above. Following this each element can be combined into a complete application.

The algorithms and other knowledge required to achieve the objectives will be gathered from the consultation of working papers, textbooks and general information to be found on the Internet. In addition, further help, information, and guidance will be assembled from conversations with the project supervisor and other leading academics in the field of image processing.

It is intended that the system will be implemented fully in the latest release of the standard Java Api, in conjunction with the extra Apis available for video processing (Java Media FrameWork), image processing (Java Advanced Imaging) and 3D modelling (Java 3D) available from Sun Microsystems.

Although Java is generally platform independent, one is required to development for a specific platform when using Java 3D. Therefore, it is my intension to ensure that my application is only fully working on the Windows operating system. Although it may be possible that the final project solution is multi-platform.

A.5 Deliverables

This list contains what the final system will consist of including the standard 3rd year project deliverables:

A.5.1 Software package(s) that perform

1. Scene / Camera calibration
2. Ball Location
3. Trajectory plotting
4. Trajectory prediction
5. Visualisation of the trajectory

A.5.2 Report (Including Thesis)

1. Details of the experimental set up of the cameras and hardware interfaces
2. User and design documentation for the software
3. Documented details of software algorithms used
4. Faults and error limits of calculations in the system

A.5.3 Oral Report

1. 15-20 minute presentation detailing the project, methods used to achieve a solution to the abstract, problems that were overcome, and possible improvements

2. Demonstration of the working system

A.6 Resources

The software will be developed for use on a Microsoft Windows platform, and so the departmental hardware will not be appropriate for this project. Therefore, all the necessary hardware for solving the actual problem, will be provided by the student. No hardware owned by anyone else is being used, nor is any to be built for this project. The hardware will consist of:

1. Two cameras, 30fps standard PAL format cameras

2. Standard desktop PC to program and run the required applications. The hardware will be also attached to this piece of hardware

3. Video capture board for cameras to attach to PC

However, the university servers, via project accounts, will be used to store backups of all software and documentation created as part of this project. These backups are intended to be created on a daily basis or after any major completed changed implementation occur (whichever occurs first). On a less frequent basis, backups of data will also be written to CD-ROM and to a portal hard-drive device.

The software required will consist of:

1. Java SDK 1.4 (free download from Sun website)

A.7 TimeTable

Week	Description of Intended Activities
1	Initial Research
2	Write Specification Document
3	Investigate & Implement Calibration Algorithms
4	Investigate & Implement Ball Locator Algorithms
	Investigate & Implement Triangulation Required To Recreate the Ball Position In 3D
5	Investigate & Implement Basic Trajectory Interpolation Techniques
6	Research Java 3D API For Elements Required To Build A Virtual World
	Create Basic Animation
7	Research Kalman Filter Principles
	Research Physics of Projectile Trajectories
8	Implement a Basic Prototype Kalman Filter
	Research Numerical Integration Techniques
9	Implement Kalman Filter for Ball Tracking
10	Write Progress Report
Xmas 1	Create a Prototype User Interface
Xmas 2	Complete User Interfaces Implementation
Xmas 3	Conduct Further Filming (For Use In Error Quantification)
Xmas 4	Analysis New Footage & Test Footage On Existing Software
11	Calculate Error Measurements From New Footage
12	Modify Kalman Filter With The Use Of Error Data
	Implement Tracking Algorithm Modifications
13	Enhance Animation of Ball Trajectory
14	Unit Testing
15	System Testing
16	General Tesing
	Implement Any Final System Modifications Required
17	Research Interesting Advanced Techniques
	(Which Could Be Used To Overcome System Limitations)
18	Write & Produce Presentation
19	Presentation
20	Presentation
Easter 1	Write Final Report
Easter 2	“
Easter 3	“
Easter 4	“
Easter 5	“
21	Check Final Report
	Implement Any Document Modifications Required
22	Hand In Report

A.8 Conclusion

The overall aim of this project is not to create a perfect system working at cricket grounds, but to take the system as far as we can, and solve as many problems in the given time. In this situation the process is more important than the final product. The project could be thought of as a proof of concept.

Bibliography Bibliography

[1] Andre Gueziec, "Tracking Pitches for Broadcast Television", http://www.trianglesoftware.com/pitch_tracking.htm, 2002.

[2] J.L. Crowley, P. Bobet & C. Schmid, "Dynamic calibration of an active stereo head", in Fourth International Conference on Computer Vision, pp. 734-739, IEEE Computer Society Press, May 11-14 1993.

[3] M.A. Sid-Ahmed & M.T. Boraie, "Dual camera calibration for 3D machine vision metrology", IEEE Transactions on Instrumentation & Measurement, vol. 39, pp. 512-516, June 1990.

[4] David Eberly, "Least-Square reduction of B-Spline curves", http://www.magic-software.com, June 2003.

[5] Nishiwaki, K. Konno & K.A. Nagashima, "The humanoid Saika that catches a thrown ball", in IEEE International Workshop on Robot & Human Communication, pp. 99-99, 1997.

[6] Neville de Mestre, The mathematics of projectiles in sport, Cambridge University Press, 2nd Ed, 2000, ISBN:0-521-39857-6.