University of Bristol

DEPARTMENT OF COMPUTER SCIENCE

Tracking Football Players using condensation

Peter Mountney

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of Master of Engineering in the Faculty of Engineering

May 2003 | CSMENG-03

Executive Summary

Aim

The aim of the project is to produce a reconstruction system which generates 2D animated goals and highlights from short video sequences of football matches.

Motivation

Broadcast rights in football are extremely expensive and tightly controlled. This has led to millions of fans using the internet to follow football matches. These fans are supplied with static photos and text commentary but the experience is relatively dull and survives only as a result of the fanatical nature of football fans. This project aims to produce a system which is capable of creating 2D animated goals and highlights for distribution across the internet, bringing back some of the excitement and spontaneity of football.

System Overview

The reconstruction process involves tracking players. Using computer vision to track human motion has established itself as a principal area of research, however, it still presents a number of interesting non trivial problems including tracking multiple objects in a congested scene with occlusion. There are also problems related specifically to tracking the motion of footballer players; including varied player motion, players' appearance altering over short periods of times and weak distinguishing features. Additional complications arise from the video capture mechanism and process.
Camera Calibration needs to be implemented in order to produce a 2D animation of the players real world positions.

Achievements

- I have written a total of 3500 lines of code in C++ and Java. A GUI has been produced using the Java Media Framework which is capable of grabbing frames from video files, viewing animated highlights and displaying various output for tracking analysis.
- A measurement model has been created as part of the CONDENSATION algorithm The model is based on blob tracking and background subtraction.
- A motion prediction model has been created as part of the CONDENSATION algorithm, which estimates motion in the world coordinate system and projects it back into the image coordinate system.
- An additional step has been added to the CONDENSATION algorithm which implemented an approach to occlusion reasoning using contextual knowledge.
- An animated goal has been successfully generated from footage of a football match.

In total over 550 hours have been spent on this project. This time was spent learning the Java Media Framework, the Java Native Interface and C++ as well as understanding the Bayesian maths used in the condensation algorithm and researching appropriate computer vision techniques.

1 Background

1.1 Problem statement

Object tracking is the problem of following the movements of a target object over a period of time[1]. In computer vision objects move relative to a camera causing the projected images of the objects in the camera plane to change.
In this thesis the tracking problem is defined as:

given an observation of an object at time t, determine the most likely location of the same object at time t+1 [3]

Figure 1

In this project players will be tracked over a short sequence footage taken from a television broadcast of a football match. Figure 1 shows an example of the footage used. This is a multiple object tracking problem which presents the following difficulties. **Occlusion**: multiple objects causes occlusion as they overlap. Occlusion occurs regularly during football matches. Both partial and full occlusion present problems as the players appearances change or disappear. **Background clutter**: a problem arises when regions in the background appear similar to the players. In football this can occur if the team colours are similar to the pitch, if there are sponsors logos painted on the pitch or the player is near the crowd such that the crowd becomes the background. **Player motion** tends to be varied and sudden due to the nature of the game. Consequently players move at variable speeds and in variable directions making them hard to track. **Players' appearances** vary significantly over short periods of times for a number of reasons including running, making a tackle or raising an arm. **Environmental change:** changes in lighting and weather affect the appearance of objects. **Camera Motion**: the capture process creates motion errors as a result of panning and zooming. Players furthest from the camera are also represented by less pixels than those close to the camera.

These problems will be addressed in this thesis.

1.2 Introduction to tracking

Tracking systems are employed for a variety of reasons including surveillance, road traffic analysis and gesture recognition. There are an equal number of approaches which tackle the problems associated with tracking.
Although these approaches are varied they share common structure and terminology.

The generic tracking algorithm consists of 4 stages

1) Initialize the target object representation

For the image at time t

2) Generate a set of candidate regions
3) Measure the likelihood that a candidate region represents the target object
4) Determine the most likely location of the target object.
   t = t+1

The **target object** is defined as the object which is being tracked. In order to track an object it must have a method of representation which distinguishes it from other objects and regions. A **candidate region** is a region in an image at t+1 which may represent a target object. The method of generating the location of candidate regions is often referred to as the **Prediction Model** as it predicts potential location for the target object.
Each candidate region is compared to the target object representation in order to measure the similarity. Tracking algorithms assess this similarity using a **Measurement Model**. The measurement model comparing the candidates region representation to the target object representation to allow the algorithm to resolve which candidate best matches the target object.

1.3 Template Matching Algorithm

Figure 2 shows a simple example of a tracking problem, a single object moving at constant velocity from the left to the right of the frame.

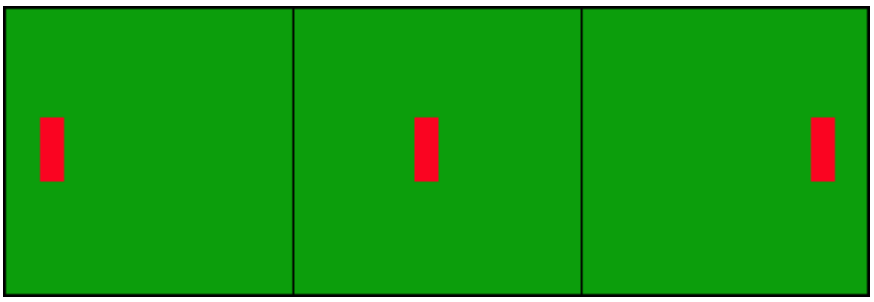


Figure 2

It is easy for humans to track the path of the red rectangle. However, this simple problem presents a number of issues that lie at the heart of the tracking problem. The following section gives an overview of a basis tracking algorithm, known as Template Matching.

The algorithm for a basic template matching algorithm is given below.

Template Matching Algorithm

- Initialize the target object (red rectangle) in t_0
- Observe and create a template representing the target object
- For t_0 to t_n
  - Place the template at all possible locations, measure for a match by calculating the SSD (sum of squared difference) of the RGB values.
  - The SSD closest to 0 gives the location of the target object at t+1.

1.4 Target object/candidate region representation

In the template matching example the target object was represented as a rectangular template consisting of 400 pixels with an RGB colour vector (255,0,0). Candidate regions for this example are generated placing a rectangular template of 400 pixels (the same size and shape as the target object representation) on the image and extracting the image data. For example if a candidate region were over the green background, the representation would consist of 400 pixels with an RGB colour vector (0,150,0).

1.5 Prediction Model

The most basic technique is used. The entire image is taken to be the *search area* and candidate regions are extracted at every possible location. This is not a computationally efficient approach and the process can be optimized by using a resolution pyramid.
More common is the use of motion models to predict the motion of the target object.

1.6 Measurement model

The Measurement Model in the Template Matching example the Measurement Model was SSD.

Template Matching is effective for the example and has been a popular method for tracking objects [4]; however, its value in non trivial tracking problems is limited by the two general assumptions:

1) "Constant Intensity" which assumes the observed brightness of any object is constant over time.
2) 2) The assumption that all the pixels in the template have the same image displacement [5]

These assumption are regularly violated in real world tracking problems.

Real world tracking problems are complex and almost impossible to solve without knowledge of the target object and the environment the in which it is being tracked

1.7 Knowledge of the Target Object and Environment

Gestalt theory suggests that perceptual organization is in essence based upon constructing simple and regular forms. This can be expressed by the following four principles: similarity (common colour and texture), continuity (smooth surface and lines), good form (symmetrical shape) and common fate (moving together)[6]. Visual Psychologists believe that perception and interpretation of images is directly related to previous experience and knowledge of the image.

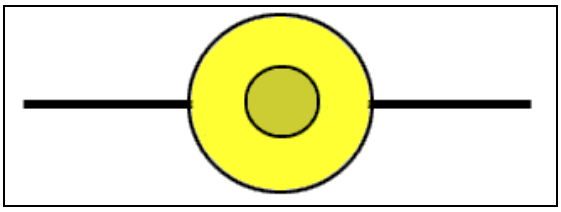


Figure 3 [7]

Without knowledge, figure 3 is hard to interpret. However with knowledge of the geometric structure of Mexican hats and bicycles, it becomes clear that this is an illustration of a man on a bike wearing a Mexican hat.

Knowledge is essential to successful tracking. It allows assumption to be made which constrain the tracking problem. The nature and correctness of these assumptions directly affects the robustness of the tracking algorithm.

In the context of this project it is known that the objects being tracked are humans involved in a game of football. It is therefore possible to make assumptions about the representation of these objects and how they deform. It is also possible to make assumptions about the motion of the players.

1.8 Overview of methods used in tracking

Knowledge of the target object its environment is used in a number of different ways. In [8] McLauchlan and Malik define four techniques which can be employed in tracking systems.

1) Model-based trackers - Target objects are represented by fixed geometric models or deformable models
2) ``Blob'' trackers - Target objects are represented as regions which can be segmented from the background.
3) Feature tracking - point and line features are used to represent the target object.
4) Motion Modelling - This approach models the motion of objects to predict likely locations at t+1.

The following section evaluates these techniques and reviews their implementation in tracking systems.

1.9 Model-based Object Representation

Knowledge of the target object's shape or appearance can be used to build an explicit target object representation. The aim of this technique is to improve the robustness of the tracking algorithm by reducing the space of possible appearance changes. There are two types of model based representations. Fixed geometric models and deformable models also knows as active contours or snakes.

1.9.1 Geometric Models

There has been a great deal of work in the field of computer vision which is related to representing objects as geometric models. The following three approaches have been used in tracking related work [2].

1.9.1.1 3D-model-based representation

Knowledge of the target object's 3D shape is used to create a 3D model which represents the target object. The 3D models usually take the form of volumetric or wire-frame models. Model-based recognition [9] [10] is then used to locate objects in images and allow them to be tracked over a sequence of images.

3D model-based tracking systems have been successfully developed for vehicle tracking. [11] [12] [9]. The models shown in figure 4 were used in [11] to track moving vehicles. These 3D models are generic and do not accurately represent vehicles however it is known that vehicles do not change shape only position, orientation and scale relative to the camera. In [8] the limitation of this a reliance on detailed geometric models. pproach are shown to be its

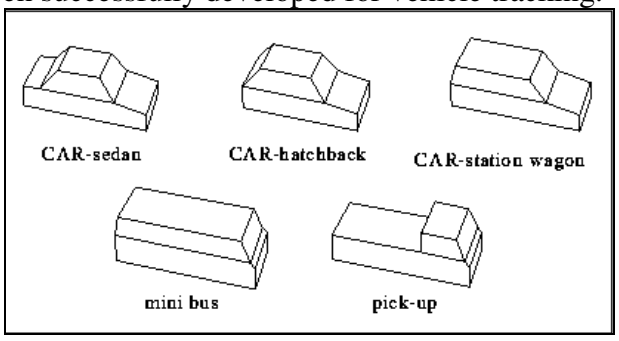


Figure 4

It is infeasible to create models for all the vehicles found on the roads.
While the 3D model based approach is suited to ridged non-deformable objects, it has been used to track human motion by interpreting the body as a series of articulated objects [13][14][15]. The work done in this area is often limited to constrained examples and requires high resolution video footage, leading to computationally expensive processing. However its use in human action recognition makes it an appealing method of object representation for tracking football players.

1.9.1.2 View-based representations

In this approach 2D models are created which represent the target object from views which are likely to appear in the frame. The models are built from knowledge of the object such as contours and extracting image information like blobs. This technique is common

tracking human motion [16][17][18] (Haritaoglu et al., 1998a). Tracking hands, bodies and silhouettes.
The main problem with view-based representations is in generating the model. In [19] a Bayesian approach is suggested where the models representing 2D human motion are learnt from a training set. This approach is still restricted by the size of ,and variance in, the training set. View based representation performance is markedly impaired when objects are viewed from unusual view-points[20].

1.9.1.3 Appearance-based representations

This approach is similar to view based representation in that it learns the appearance of the object through a set of training images which show the object from various views. The method in which the object is represented is very different as it employs techniques originating in eigenimages. [21]Eigenspace is produced using Principal Component Analysis. The Eigenspace represents most of the variations in the images in the training set. This approach has been used in hand tracking [8] [22] [23] and face tracking [24]

The benefit of appearance-based representations is that the representation can be acquired through an automatic learning phase [25] which is not the case with traditional shape representations.
However, the limitations of this approach are its inability to deal with occlusion or segmentation. The robustness of this approach is dependent on the training data used to generate the eigenspace. If a change in appearance due to a factor such as a change in lighting has not been seen in the learning process, tracking will fail.

1.9.2 Deformable Models

1.9.2.1 Active Contour Based Tracking

Active shape models is a generic term which covers snakes, deformable templates and dynamic contours. In these approaches the target object is represented by the bounding contour of the object. The representation is dynamically updated at each image in the sequence.
In [7] Blake and Issard describe active shape modelling as using prior knowledge to impose constraints on the object's properties, such as continuity and smoothness, from the start (these properties usually emerge from the image data). An elastic model of a continuous flexible curve is imposed upon and matched to an image. By varying the elastic parameter, the strength of prior assumptions can be controlled.
Previous work in using active shape models for tracking include vehicle tracking [26][27] and in tracking human motion[88].

Active contours have been used to extract the entire outline of the humans [28] and individual body parts [71]. Rigoll et al [29]propose a stochastic approach to silhouette extraction incorporating pseudo-2D hidden Markov models (HMM)to produce a Discrete Cosine Transformed (DCT) representation of the image.

Contour based approaches are computationally more efficient than region based representation. However active shape modeling does not deal well with occlusion and requires very specific occlusion reasoning. In [8] Koller *et al* use an occlusion reasoning approach based on Kalman Snakes to track multiple vehicles. This approach works well but only in a highly constrained environment.

The shortcomings of purely contour-based approaches is that they neglect information contained within the body of an object, focusing purely on its external area [30]. This is not suited to tracking multiple objects with similar shapes such as humans.

1.9.3 Blob trackers

In [31] blob like entities are described as being formed by the grouping together atomic parts of a scene based on proximity and visual continuity.
Blob tracking defines the target object as one which can be segmented from the rest of the scene using a foreground/background algorithm[26]. There are various approaches to extracting blobs [32] [33]

In general the foreground/background algorithm is initialized by the background subtraction technique. The background model is an estimation which represents what the scene would look like if there are no moving objects in the scene. Background models can either be static or more commonly adaptive [34] to allow the model to evolve compensating for changes in lighting and weather.
Foreground objects are detected by subtracting each new image from the background model, pixels of interest are identified by thresholding the difference. The result is moving objects are identified as foreground. Several approaches to tracking extend background subrataction to use multivariante Gaussians [31] or Gaussian mixtures[35][36]

For tracking single objects this technique works well as a single blob is produces which can be evaluated to identify its centre [31]. The technique is extendable to tracking multiple objects and it is used successfully by Coifman [37] to track road traffic. However, he concluded that while the technique works well for free flowing traffic as regions of interest are spatially isolated producing distinct foreground blobs, the technique is less effective when the traffic is congested.
Traffic congestion causes occlusion and congestion in the image space making the segmentation of individual vehicles difficult. The result is that segmentation produces large foreground blobs which represent several vehicles.
In [38] McKenna et al address the issue of occlusion and congestion in segmentation. Segmentation is extended from separating background and foreground to identifying three levels of abstraction - regions, people and groups. This approach uses an adaptive background subtraction method that combines colour and gradient information. The

system was run in several different indoor and outdoor scenarios successfully tracking people however it consistently failed when two people clothed in a similar manner come together to form a group or move apart, splitting a group.

The effectiveness of Blob tracking is limited by the tracking environment. It requires that a background model can be generated which accurately estimates the scene without moving objects. This is not always possible if there is camera motion.
This technique is suited to the tracking of football players as the background is predominantly green allowing good segmentation.

1.9.4 Feature Based Tracking

Feature based trackers can be divided into two groups in [43] according to the type of features tracked.

1.9.4.1 Correspondence-based techniques

In this technique features are extracted from two sequential images I_t and I_{t+1} [44]. These features are typically lines, edges or corners. Feature matching is performed in the image plane to establish correspondence between the features in I_t and I_{t+1}.
The fundamental advantage of this technique is that is does not represent the whole of the target object. Instead a set of sub-features is used as a representation.

This allows for robust tracking even when the target object is partially occluded. It is also robust to changes in lighting as features such as corners are in general not significantly affected by changes in the environment. This approach is only effective if the target object can be represented as a set of sub-features which are either lines, edges or corners. The main problem with this approach is that when the feature matching fails, the correspondence error is significantly large.

1.9.4.2 Texture correlation-based

Template matching as described earlier is a simple and oldest [45] approach to recognizing an object in an image. It often fails when the tracked object's features deform or changes over time. However, this approach formed the starting point for approaches such as deformable templates and subsequently active contours.

1.9.5 Motion Modelling

Tracking systems often employ prediction models which model and estimate motion in order to reduce the search area and increase computational efficiency. Based on previous positions and the characteristics of the target object's motion it is possible to predict the position of the object at t+1.
Basic motion predictors are based on constant velocity or constant acceleration[39]. Specific motion models have been used to predict human motion such as walking [40].
Tracking systems which require more sophisticated motion models commonly use optical flow.
Optical flow based techniques [41] aim to obtain vectors that estimate the motion of the gradient within an image sequence. The intention is to used this vector to interpolate the 3D motion of the target object to reduce the search area. Okada et al used prior knowledge of the human shape to interpret motion estimation to successful implement a single person tracker in real time. This technique is limited as it requires specialized hardware to operate at real time and optical flow can only be found for textured area. Multiple objects have been tracked [42] using optical flow; however, it is not effective for occlusion or cluttered areas. Motion modelling is also used within the frameworks of Kalman Filtering and Condensation.

1.10 Tracking Systems

Typically tracking systems will employ several of the techniques described in order to solve the tracking problem. There is no single tracking system which can be generically applied to all tracking scenarios. However there has been wide success with Kalman Filtering and the Condensation algorithm. These are probabilistic approaches to tracking which have been successfully applied in various circumstances. The following section presents a review of these approaches.

1.10.1 Probabilistic framework

In a probabilistic [46] approach the states of the target object and the object representation are taken to be X and Z respectively. In a dynamic system the states and measurements are denoted by X_t and Z_t at time t.
The tracking problem can be formulated as an inference problem with the prior $P(X_{t+1}| Z_t)$ which is a prediction density. Further from this it is possible to represent the measurement/observation likelihood:

$$P(X_{t+1}| Z_{t+1}) \quad \alpha \quad P(Z_{t+1}| X_{t+1})\, P(X_{t+1}| Z_t)$$

and the dynamic model (prediction model):

$$P(X_{t+1}| Z_t) \quad = \quad \int_{xt-1} P(X_{t=1} \mid X_{t-1})\, P(X_t \mid Z_t)$$

Tracking can be taken to be the probability density propagation from $P(X| Z_t)$ to $P(X_{t+1}| Z_{t+1})$. Therefore the aim of the tracker is to determine the probability density for the target's state at each time-step t [47].

The propagation is a three stage process
Deterministic – The deterministic element of the prediction model causing the drift of the density function.
Stochastic – The stochastic element of the prediction model which models uncertainty causing a spreading in the density function
Reactive reinforcement due to measurements – Observations made at time t, are used to reinforcement of the predicted density in the regions close to the observation.

If it can be assumed that all distributions are Gaussian it is possible to use the mean and covariance to parameterize the probability densities. Consequently the probability density propagation updates these parameters. The Kalman Filter is a technique which estimates a set of optimal probabilistic parameters.

1.11 Kalman filtering

The Kalman filter [48] [49] [50]is a recursive algorithm which provides Minimum Mean Square Error (MMSE) estimations of the target object's position and uncertainty in the new frames. Essentially the algorithm determines a predicted position and subsequently a search region.

1.11.1 Mathematical Methods

Kalman filtering assumes

1) A Linear state model
2) The uncertainty is Gaussian with zero mean

The target object's position and velocity at time t are represented as

$$p_t = (x_t, y_t)$$
$$v_t = (v_{x,t}, v_{y,t})$$

The object's state at t is represented by its position and velocity

$$s_t = [x_t, y_t, v_{x,t}, v_{y,t}]^t$$

The aim of Kalman filter is to compute the state vector for subsequent frames.

$$\textit{Given } s_t \textit{ computer } s_{t+1} .$$

Kalman filtering consists of two parts state predication (computed using the state model) and state updating (computed using measurement model).

Dynamic State model (Prediction Model)
The state model describes the temporal part of the system. Kalman filtering constrains the state model to be linear such that:

$$s_{t+1} = \Phi s_t + w_t$$

where Φ is the state transition matrix (deterministic) and stochastic element w_t has a zero mean, normal Gaussian distribution: $w_t \sim N(0,Q)$.

Measurement model.
The measurement model must be linear:

$$z_t = Hs_t + v_t$$

where H relates current state to current measurement and v_t represents measurement uncertainty which is normally distributed as zero mean Gaussian $v_t \sim N(0, R)$.

1.11.2 Kalman Filter Algorithm[51]

The Kalman filter is a recursive process with the following four steps:

1) **State and covariance prediction**
The current state s_t and its covariance matrix Σ_t are known.
State prediction involves two steps:
1. State predication: $s^-_{t+1} = \Phi s_t$
2. Covariance prediction; $\Sigma^-_{t+1} = \Phi\Sigma_t \Phi^t + Q$

2) **Measurement to obtain** z_{t+1}
Using the measurement model searches for the region determined by the covariance matrix Σ^-_{t+1} to find the target object at time t+1, z_{t+1}. The search region which contains the actual state with a given probability c^2 is an ellipse, and satisfied the following equation:

$$(p-p^-_{t+1})\,(\Sigma_{t+1}{}^{p-})^{-1}(p-p^-_{t+1})^T \leq c^2 \qquad (c=0.95)$$

3) **Computing gain matrix**
The gain matrix K is a weighting factor that determines the contribution of the measurement z_{t+1} and the predication Hs_{t+1} to the posterior state estimate s_{t+1}. Gain matrix can be computed by the following formula:

$$K_{t+1} = \Sigma^-_{t+1} H^T (H\Sigma^-_{t+1} H^T + R)^{-1}$$

4) **Posterior state and covariance estimation**
The posterior state estimation is the combination of the state predication s^-_{t+1} and the measurement z_{t+1} :

$$s_{t+1} = s^-_{t+1} + K_{t+1}(z_{t+1} - Hs^-_{t+1})$$

The posterior covariance estimation can be obtained by:

$$\Sigma_{t+1} = (I - K_{t+1}H)\Sigma^{-}_{t+1}$$

At each time step the Kalman filter recursively conditions current estimate based on all of the past measurements. This process is repeated using the previous posterior estimates as the new prior estimate.

1.11.3 Limitations of Kalman Filtering

The process and observation noise processes in real world problems cannot be white noise and the noise covariance matrices Q and R are commonly only known to within an order of magnitude. However, violating the conditions imposed by the Kalman filter does not render the algorithm useless although its performance will be affected. The algorithm's estimates, for the target object's position and velocity, may not meet the MMSE criterion.

Despite the limitations, Kalman filtering is a widely used algorithm which forms the basis of many successful tracking systems. [52][53] Employ Kalman filters in the tracking of humans for the purpose of pedestrian surveillance and pose recognition.

The main downfall of Kalman filtering is its Gaussian representation of probability density. This representation is essentially uni-modal and consequently it can only support one hypothesis for the state of the target object at any given time t. When tracking objects through cluttered backgrounds the Kalman filter can loose track of the target object as it locks onto background features.

1.12 The CONDENSATION algorithm

The CONDENSATION algorithm (Conditional Density Propagation) [54][55][56] allows the probability density representation to be multi-modal. As a result it is capable of simultaneously maintaining multiple hypotheses about the state of the target object. This means that tracking systems based on condensation can be made robust as they can recover from temporary ambiguities arising from background features appearing more like the target object representation than the target object itself.
The recovery process occurs over subsequent time steps by rewarding or providing reinforcement for the hypothesis which represent the target object and by punishing the hypotheses which represent the background or other objects causing these hypotheses to gradually diminish.

In addition to the algorithms robustness in cluttered areas, the Condensation algorithm also allows the use of non-linear prediction models which are more complex than those commonly used in Kalman filters.

1.12.1 Mathematical Methods

The Mathematical methods [57] are broken down into the following sections:

The *a posteriori* measurement density: $P(x_t \mid Z_t)$
The *a priori* measurement density $P(x_t \mid Z_{t-1})$
The process density describing the dynamics $P(x_t \mid x_{t-1})$
The observation density $P(z_t \mid x_t)$

Probability Distribution: The target object has a state vector $x \in \boldsymbol{X}$. It is assumed that the exact state of the object cannot be known. What is known about the objects is described using a probability function P(x).

Prediction model: As the observed scene changes over time, the probability function evolves to represent the altered object states.
A stochastic differential equation is used to describe the dynamics of the evolution. The equation comprises two elements: the deterministic and stochastic.
The density function $P(x_t)$ depends only on the immediately preceding distribution $P(x_{t-1})$.
Therefore the process density describing the dynamics is determined by $P(x_t \mid x_{t-1})$.

Measurement Model:
Let z_t be the measurement at time t with history $Z_t = \{z_0, \ldots, z_t\}$.
Consequently the *a priori* density can be represented as $P(x_t \mid Z_{t-1})$ and the *a posteriori* density as $P(x_t \mid Z_t)$.

Based on the assumption that the measurements are independent it is possible to calculate the *a posteriori* density $P(x_t \mid Z_t) = P(x_t \mid z_t, Z_{t-1})$ using Bayes' rule:

$$P(x_t \mid Z_t) = \frac{P(x_t \mid x_t, Z_{t-1})\ P(x_t \mid Z_{t-1})}{P(z_t \mid Z_{t-1})}$$

$$= \quad k\, P(z_t \mid x_t, Z_{t-1})\, P(x_t \mid Z_{t-1})$$
$$= \quad k\, P(z_t \mid x_t)\, P(x_t \mid Z_{t-1})$$

k is a normalization factor

The a priori density $P(x_t \mid Z_{t-1})$ is produced by applying the prediction model to the *a posteriori* density $P(x_{t-1} \mid Z_{t-1})$ of the previous time step:

$$P(x_t \mid Z_{t-1}) \quad = \quad \int_{xt-1} P(x_t \mid x_{t-1})\, P(x_{t-1} \mid Z_{t-1})$$

The complete tracking scheme first calculates the *a priori* $P(x_t \mid Z_{t-1})$ density using the dynamic model and then evaluates the *a posteriori* density $P(x_t \mid Z_t)$ given the measurements:

$$P(x_{t-1} \mid Z_{t-1}) \quad \text{—}dynamics\text{→} \quad P(x_t \mid Z_{t-1}) \quad \text{—}measurement\text{→} \quad P(x_t \mid Z_t)$$

Factored Sampling:
Sampling is used to represent distribution because typically the *a posteriori* density $P(x_t \mid Z_t)$ is too complex to be simply evaluated in closed form. In addition the state space **X** is multidimensional and significantly larger such that it is not possible to sample $P(x_t \mid Z_t)$ at regular intervals. As a result an iterative sampling technique is used.

The factored sampling is provides an approximation to a probability density

$$f(x) = f_2(x)\, f_1(x), \quad x \in \boldsymbol{X}.$$

A set of samples $\{s^{(1)},\ldots, s^{(N)}\}$ with $s^{(n)} \in \boldsymbol{X}$ is drawn randomly from f(x). By choosing a sample $s^{(f)}$ with probability

$$\Pi^{(j)} = \frac{f_2(s^{(j)})}{\sum^{N}_{1} f_2(s^{(j)})} \qquad j = \{1,\ldots.N\}$$

from the sample set **s**, a new sample set **s′** is calculated. Its distribution tends to that of f(x), as $\boldsymbol{N} \rightarrow \infty$.

1.12.2 The CONDENSATION Algorithm

Further details of the algorithm can be found in [58][59][60].

Iterate

1) **Select a sample set**
Select a sample set $\{s_t\}$ representing the *a posteriori* density $P(x_{t-1} \mid Z_{t-1})$ from the previous time step.

$$s_t^{(n)} = s_{t-1}^{(j)} \text{ with probability } \Pi_{t-1}^{(j)}$$

2) **Prediction Model**
Propagate $\{s_t\}$ to obtain a new sample set $\{s_t'\}$ according to the prediction model, $\{s_t'\}$ describes the *a priori* density $P(x_t \mid Z_{t-1})$.
Propagate each sample from the set $\{s_t\}$ is accomplished by using a linear stochastic differential equation of the form

$$s_t'^{(n)} = A\, s_t^{(n)}.+B\, w_t^{(n)}$$

where $w_t^{(n)}$ is a vector of standard normal variables and BB^T is the process noise covariance.

3) **Measurement Model**
Measure and weight the new positions in terms of the measurement feature Z_t:

$$\Pi_t^{(n)} = P(Z_t \mid X_t = s_t'^{(n)})$$

The Condensation algorithm has several advantages over Kalman Filtering: not least is its low computational cost and greater potential for real-time use. The algorithm's main advantage is its ability to maintaining multiple hypotheses about a target object. The Condensation algorithm has been successfully implemented in a number of tracking systems to track cars [61] and track animal motion [62].

1.13 Previous work in tracking Football Players

Tracking football players is still a relatively new application of tracking. It is only with the advance of hardware in the 1990's that attempts have been made to tackle this problem. The motivation has been very different in each case. This section will review the prominent works conducted in this area.

1.13.1 A video-Based 3D-Reconstruction of Soccer Games

by Thomas Bedie 2000 [63]

The motivation behind this work was to produce a system which is capable of reconstructing 3D animations of parts of a football match in order to enhance sports coverage on television. This approach does not use the condensation algorithm: tracking the trajectory of the ball and players is done manually by marking up key frames.
Regions of interest are extracted using a blob segmentation approach. These regions are texture mapped onto rectangles in 3D space. Bedie uses multiple camera angles of the same sequence of images and camera calibration is achieved using two cameras.
The results of this project are good: however, it does not deal with issues of tracking as it is a highly manual approach. This system is an example of the typical approach most commercial sports analysis products employ[68]. In these systems camera calibration is a priority and tracking is a manually intensive process which is perceived as financially cheaper.

1.13.2 Closed-World Tracking

by Intille et al [64][65][66][67]

The motivation behind the work by Intille et al is to produce a video annotation and action recognition for American Football. In this series of papers the concept of closed world tracking is defined and the importance of context is raised.
A closed-world is defined as a region of space and time in which the specific context is adequate to determine all possible objects present in that region. Bounding boxes are placed around players at the start of each play. These boxes represent closed worlds. This approach makes use of knowledge about the rules of American football and likely events. This contextual knowledge can be used to resolve some tracking ambiguities.
This approach successfully tracks American football players over several plays including situations when players occlude each other and collide. This is one of the most successful sports tracking systems. The approach has been applied to other sports [89] with limited success. This is due to the lack of contextual knowledge it is possible establish for other sports.

1.13.3 Where are the ball and players? : Soccer Game Analysis with Color-based Tracking and Image Mosaick

by Seo et al [69]

The motivation behind this work is to compute the locations of football players and the ball over a short piece of footage for the purpose of match analysis.
Background subtraction is employed to identify regions of interest based on RGB color information. The players and ball are tracked by template matching and Kalman filtering. An attempt at occlusion reasoning is made by using colour histogram back-projection; however, Seo et al only consider occlusion between different teams.
To find the location of a player in real world coordinates, a field model is constructed and a transformation between the input image and the field model is computed using feature points when the centre circle is visible. Otherwise, an image-based mosaicking technique is applied. By this image-to-model transformation, the absolute position and the whole trajectory on the field model is determined.
This work shows successful multiple player tracking. It proposes a simple approach to specific types of occlusion. However this approach is neither robust or extendable.

1.13.4 Tracking multiple sports players through occlusion, congestion and scale

by Chris Needham [70]

The motivation for this work is to produce a system capable of positional behavioural analysis. In this work the condensation algorithm has been used to track up to six players on an indoor five-a-side football pitch. Image segmentation is achieved by creating prior colour space models built offline for foreground and background or player and non player. The models use HSI space as it produces greater separation between foreground and background. For each pixel in the image the probability that it is foreground is computed using Mahalanobis distance. Segmentation is improved with probabilistic relaxation. Finding the corresponding player at t+1 is done by simply fitting a bounding box around each silhouette extracted by the image segmentation and evaluating how well this fits the image data.
The prediction model used is taken from "visual tracking using closed worlds" S.S Intille and A.F Bobick.

$$x_t^i = x_{t-1}^i + \xi_x \qquad y_t^i = y_{t-1}^i + \xi_y$$

for $i - \{1,\ldots,n_j\}$ and
$\xi_{x \text{ and}} \xi_y \sim N(0,\sigma)$

To further control the samples and prevent them from straying, a Kalman filter has been added. Needham presents a basic framework to tracking football players. The results produced by this work show that players can be tracked in a controlled environment; however, there are still a number of issues which this approach does not address. Needham controls the environment in which the framework is implemented by using footage which minimise the occurrences of congestion or occlusion. Any occlusion that does occur is between players with different shirt colours. This approach does not fully tackle the issue of occlusion or the possibility of a target object being tracked more than once.

2 Technical Basis

2.1 Approach

A Condensation based approach has been chosen for this project for its ability to run in real time and its capability to represent multi modal measurement distribution and consequently maintain multiple hypothesis. Objects are represented by blobs which are extracted from the image data using background subtraction. The objects' motions are estimated by a prediction model which uses the world coordinate system not the image coordinate system. The deterministic component of the prediction model is a first order auto-regressive (constant velocity) motion model. The stochastic component has Gaussian distribution. An approach to clutter and occlusion reasoning is presented which uses contextual knowledge.

Tracking multiple objects with condensation can take two approaches. Using a single tracker to track multiple objects or using multiple tracker which track single objects. Neither approach is free of complicated problems however when dealing with congested areas and occlusion using multiple trackers can often result in multiple trackers tracking the same object. A single tracker algorithm has been implemented as it is more suited to the image data. Within the context of the condensation algorithm the samples are divided equally between each object being tracked. Each particle has an identification number associated to it. This is used to identify which object the particle is intended to track This allows individual object representations to be used.

2.2 Key Problems

The Key problems with tracking football players are reiterated below:

- Background clutter. A problem arises when regions in the background appear similar to the object representation. In football this can occur if the team colours are similar to the pitch, if there are sponsors logos painted on the pitch or the player is near the crowd such that the crowd becomes the background.

- Weak distinguishing image features. Football players on the same team wear matching kit. Although the kits of the two teams will be distinguishable by colour, players on the same team have few features which individualize them. It is only the face, skin colour and the name and number on the back of the shirt which can be used to tell them apart. These are not dominant features in low resolution image data.

- Tracking multiple objects is challenging, with errors resulting from sample impoverishment and probability decay.

- Tracking multiple objects causes occlusion as objects overlap. Occlusion occurs regularly during football matches. Both partial and full occlusion present the problem that an accurate match for the target object does not exist. Occlusion can result in sample depletion or multiple trackers following the same object .

- Player motion tends to be varied and sudden due to the nature of the game. Consequently players move at variable speeds and in variable directions.

- Players' silhouettes and appearances vary significantly over short periods of times for a number of reasons including running, making a tackle or raising an arm.

- Camera Motion. The capture process creates motion errors as a result of panning and zooming.

- Players furthest from the camera are represented by less pixels than those close to the camera.

2.3 Reconstruction System Overview

Figure 5 shows an overview of the reconstruction system which generates the 2D animated goals from video footage of a football match.

The system is broken down into two phases: the initialisation phase and the tracking phase. The initialisation phase occurs only once when the reconstruction system is started. The tracking phase is diagrammed in the lower half of figure 5. This diagram represents the steps involved in tracking players. The grey boxes represent the condensation algorithm. The red boxes represent additional steps which have been added to the algorithm to deal with tracking multiple objects and occlusion.

In the sections which follow, each of the steps of the system will be analysed. The problems which are associated with each step will be broken down and an explanation of the solution implemented will be given.

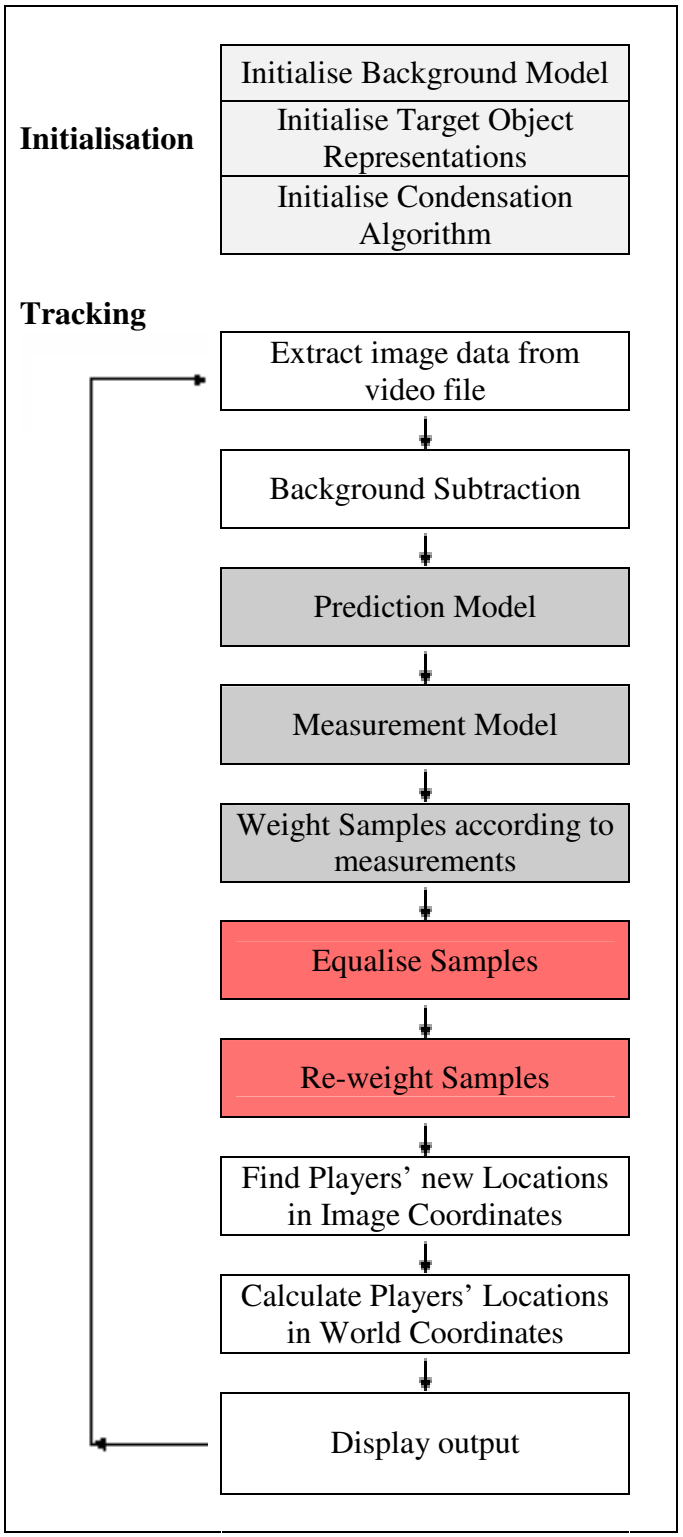


Figure 5

2.4 Image Data

The characteristics of the image data are the dominant factors influencing the construction of the measurement model and object representation.

In this instance, the image data is footage taken from television broadcasts of a football match. As a result this footage is high resolution. This presents an enormous amount of information that can be used to track the football players. With this amount of data and in light of work carried out on modelling humans, a possible solution may incorporate the use a geometric model to represent the football players.
There are two immediate problems with this approach. Firstly, using high resolution footage will be computationally expensive; secondly, geometric modelling is more suited to gesture and action recognition. It is unnecessarily complicated for the purpose of identifying a player's real world coordinates.

Figure 6

For computational reasons low resolution footage has been used. Figure 6 is an example of footage used. The resolution is 342x278.
This footage is produced by compressing high resolution footage and it is therefore subject to compression artefacts. [72] Compression techniques produce adequate results hence compression artefacts are small errors which are dealt with by accurate object representation and robust measurement models.

Blob tracking has been selected as a means of target object representation. This approach best fits the image data. Background/foreground segmentation can be easily achieved as the pitch is relatively constant. The players silhouettes vary significantly over short periods of times and blob tracking has been shown to effectively track objects of this nature.

2.5 Background Model

2.5.1 Establishing the tracking area

Before it is possible to extract regions of interest, it is necessary to constrain the tracking environment by establish the tracking area. The tracking area is the image area in which the tracking algorithm is implemented. The area is defined in the image plane by manually marking up areas which the player will not enter or areas in which it is not feasible to track. The stand and crowd are marked up producing a tracking area comprising the pitch. Defining a tracking area prevents problems arising from samples being leading to misclassification. thrown into

Figure 7

the crowd It also prevents background clutter. The tracking area reduces errors in background subtraction as it eliminates dynamic background changes produced by the crowd and creates a background consisting mainly of pitch with a small variation in colour space.

2.5.2 Background Subtraction

Background subtraction is a common method for real-time segmentation of moving regions in image sequences. The background model is the estimate of what the image would look like if there were no moving objects in it.
Background subtraction is performed by thresholding the error between the background and the current image.

2.5.3 Static Background Model

Static background subtraction requires the manual initialisation of a single background model. This model is then used with the image at time t to perform background subtraction. It assumes that the camera is static and the background is constant. Static Background Subtraction is efficient but minor changes in the background can cause errors in image segmentation.

2.5.4 Adaptive Background Model

Adaptive Background Subtraction has developed out of a need to deal with problems created when the background model changes significantly over time causing static background subtraction to identify changes in the background as motion. Segmentation errors can occur as a result of changes in lighting or weather conditions, shadows and other arbitrary changes to the background image.

Previous work has been done in the area of Adaptive Background Subtraction: Pfinder uses multivariant Gaussians [73] and Gaussian Mixture Models have been used in [74] and in [75]combined with gradient information.
These techniques are based on the assumption that background changes are slow relative to the motion of tracked objects.

The motion of players on a football pitch violates this assumption. For significant periods of time a goalkeeper's motion is slow relative to the motion of other tracked objects. This can lead to the goalkeeper being incorporated into the background model. Outfield players will also become incorporated into the background model should they take up tactical positions which are static or if motion is reduced as a result of the ball going out of play.

To prevent objects with slow motion becoming incorporated into the background model, static background subtraction is used. The problems addressed by Adaptive Background Subtraction such as changes in lighting were found to be trivial as changes in the background model were small and could be dealt with in static background subtraction.

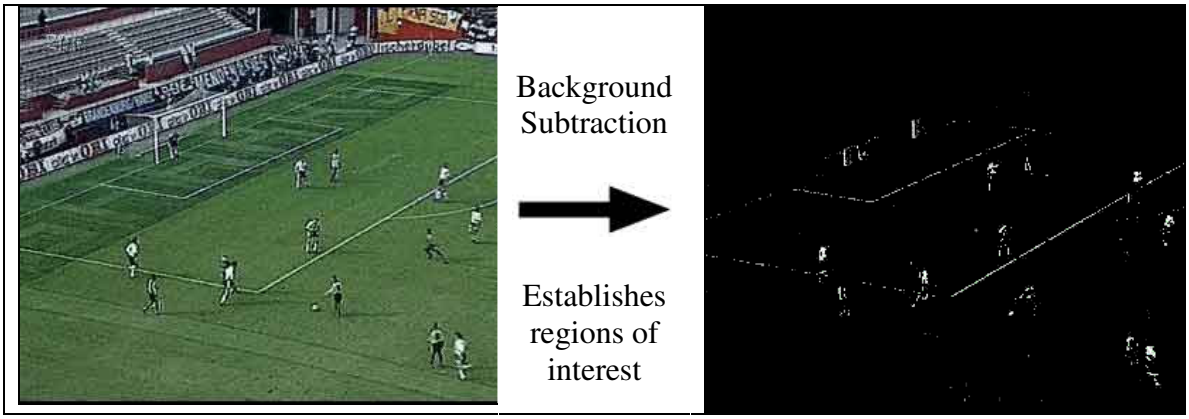


Figure 8

The static background model produced good segmentation even with camera movement. Experiments were carried out using several colour models including RGB (Red, Green, Blue), HLS (Hue, Lightness, Saturation) and HSI (Hue, Saturation, Intensity). However, the best segmentation was found by using an HSV (Hue, Saturation, Value) colour model. HSV (also known as HSB (Hue, Saturation, Brightness)) developed by Smith [76] is based on tint, shade and tone. The model is shown in figure 9

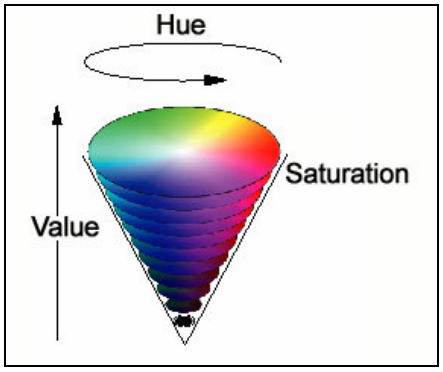


Figure 9 [77]

2.6 Object Representation

Once "blobs" or regions of interest have been extracted a method of representing these blobs is required. Players are modelled by maintaining a rectangle which represents the colour distribution of the player.

Figure 10

The RGB colour information from the rectangular box is taken to represent the player.

Individual object representation is used. At t_0 players are initialised by manually placing a bounding box around the blob which represents the player.
This is aimed at improving the results of the measurement model. In this case the major differences between object representation are as a result of the teams colours and the referee. This form of initialisation helps create accurate object representations even when objects further away from the camera are represented by less pixels than those close to the camera.

2.7 Measurement Model

At each stage of the condensation algorithm measurements are taken which assess the similarity of parts of the input image to the blobs representing players.
This measurement is done by dividing the rectangle which represents the player, R, into two halves. The top half , R_u, represents the players upper body and the lower half , R_l, represent the player's legs. The reason for this is to reduce partial representation as seen in figure 11. Partial representation occurs when regions of R are similar.

Figure 10

Figure 11

Similar areas in R leads to the candidate region Z appearing to be similar to R when it is not an accurate match. In figure 11 the white shorts of the player are similar to the white shirt. Consequently the chest area in R is similar to the shorts in Z. The players left leg in Z also corresponds to the arm in R.

Measuring the similarity between the observed data and the object representation is done using the equation:

$$(m_u * m_l) \sum_{x,y}^{N} \frac{c}{R(x,y) - Z(x,y)}$$

Where R(x,y) is the sum of the RGB values of the object representation at coordinates (x,y). Z(x,y) is the corresponding value in the candidate region.

m_u and m_l are the frequency of $R(x,y) - Z(x,y) < \alpha$. (In this work $\alpha = 10$).

The element of the equation ($m_u * m_l$) acts as a discriminator. Partial matches are punished as either m_u or m_l will be small whilst accurate matches are boosted as m_u and m_l will be large.

The measurement model has the advantage of being computationally simple. Nonetheless it provides a good metric for similarity whilst allowing the target object to deform. The target object deforms as players' silhouettes vary. The variation in shape is usually a result of limbs moving although it can be a result of players falling over. The measurement model copes well with limb movement as these are not dominant features of the object representation. The objects appearance also alters as a result objects further away from the camera being represented by less pixels than those close to the camera. If the player moves from the far side of the pitch to the near side the player will appear larger and consist of more pixels. However these changes are small and the measurement model was found to be robust to such changes in size.

2.8 Prediction Model

The prediction model comprises a deterministic element which models human motion and a stochastic element which deals with uncertainty in motion.
Player motion tends to be varied and sudden due to the nature of the game. Consequently players move at variable speeds and in variable directions. This makes it difficult to predict their motion. Many approaches attempt to model human motion by modelling the movement of parts of the body in order to predict local motion. These approaches are complex and often require exemplars to learn from in addition to being computationally expensive at execution time. Such approaches are therefore not well suited to humans running in varied direction with low resolution image data.
Commonly motion is predicted using Gaussian distribution in the image coordinate system. However, as an alternative, an approach which predicts motion in the world coordinates system has been implemented.
The previous positions of particles are kept in a history by the condensation algorithm. These previous positions are used to calculate the real world velocity of the player. The assumption is made that players move at constant velocity. Whilst this is an unrealistic assumption it can be made because the equation contains a stochastic element. The stochastic element models the unpredictable factor of the player motion. This is done using a Gaussian distribution.
The prediction model is:

$$x_t^i = |x_{t-1}^i - x_{t-2}^i| + \xi_x \qquad y_t^i = |y_{t-1}^i - y_{t-2}^i| + \xi_y$$

where i is the particle at time t. $|x_{t-1}^i - x_{t-2}^i|$ is the velocity in the x direction and ξ_x and ξ_x are the Gaussian distribution where $\xi_x \sim N(x_{t-1}^i - x_{t-2}^i, \sigma)$ and $\xi_y \sim N(y_{t-1}^i - y_{t-2}^i, \sigma)$

Whilst the stochastic element models uncertainty, it can be constrained. In athletics 100m runners travel at a velocity of around 10 metres per second. If a frame rate of 25 frames per second is assumed then these athletes move at 0.4 metres per frame. If a player falls or is tripped the motion will experience extreme deceleration. Alternatively it is not possible for a player to accelerate beyond aproximatly 0.2 metres per frame. Through experimentation it was found that a reasonable value for σ is 0.27; however, football pitch use imperial measurements (yards). Therefore σ is taken to be 0.3.

Predicting motion in world coordinates has significant advantages over prediction in image coordinates. In figures 12 and 13 dots represent the players previous positions from which the red dot is predicted by means of constant velocity.

The yellow circle in Figure 12 and the yellow ellipse in figure 13 show the Gaussian distribution of the stochastic element of the equation. Figure 12 shows the result of predicting using image coordinates (IC). Figure 13 shows the results of predicting using world coordinates (WC) and projecting the distribution back into the image plane. The shapes of the two distributions are different although they are created using the same Gaussian distribution. Figure 14 shows the contrast in shapes. This highlights the redundancy and short comings of prediction which use image coordinates.

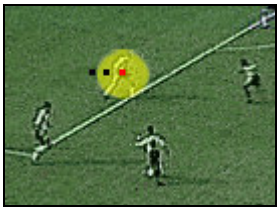


Figure 12

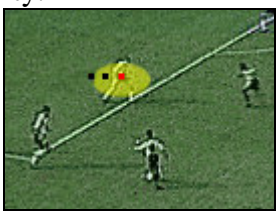


Figure 13

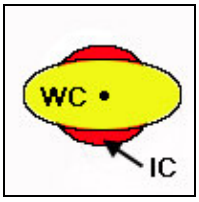


Figure 14

The prediction model use here concentrates particles in areas which the player is physically able to occupy.

Using world coordinates system for prediction not only helps model uncertainty better it also provides more comprehensive deterministic modelling. For example a player moving towards or away from the camera may not exhibit significant motion in the image coordinate system however by using the world coordinate system and the players velocity it is possible to model the players velocity more accurately.

The prediction model assumes there is only 2D motion on the plane of the pitch. Therefore no allowance is made for players jumping to head the ball. This is beyond the scope of this project as motion perpendicular to the pitch requires the recognition of the action of jumping.

2.9 Equalising samples

Equalising samples [78] deals with the problems of sample impoverishment and probability decay.

In the condensation algorithm derived particles are generated in proportion to the observed measurements. This is because in theory, modes represent potential objects. It is therefore intuitive that by generating more particles around the strong modes it is more likely to find good candidate representations of the target object.
The disadvantage of this approach is that modes will be lost.

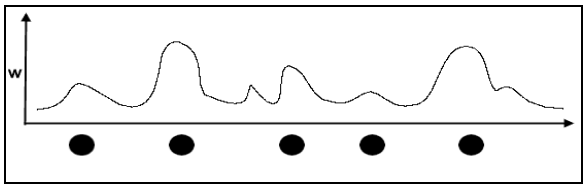


Figure 15

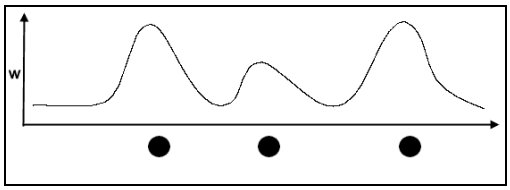


Figure 16

Figure 15 shows a 1D distribution of weights where the black circles represent players location. By generating particles proportional to observed measurement the distribution is likely to change to the representation shown in figure 16. Two of the players are no longer represented. This is known as sample impoverishment [78].

The generation of derived particles needs to be controlled to make sure that the distribution is related to the number of objects being tracked.
This is done by using importance sampling [79]. Importance sampling allows additional samples to be derived in specific areas. In Tweed's implementation an importance function f is defined. The number of samples derived from the sample set $\{s_t\}$ is proportional to f. The weight of samples derived by importance sampling is $\Pi_t/f(s_t)$. However the effect of this technique is limited as the larger the difference between Π_t and $f(s_t)$, the lower the weight of the final particle.

Probability decay occurs as a result using individual measurement models. In the idealized case of two players A and B where p and q are their weights respectively. If these weights are consistently produced over time then after t time steps the ratio of weights will be $(p/q)^t$. The weights have decayed geometrically. Probability decay can be dealt with by periodically normalizing the confidence of the hypothesis which have sufficient weights [78].

To deal with sample impoverishment and probability decay, Tweed locates clusters in the state density and constructs a Voronoi tesselation [80] based upon these cluster centres. Each Voronoi cell represents the distribution describing for the most part one object.
The following steps form sample equalization:

- At each time step, build an importance function which results in equal numbers of samples being taken in each Voronoi cell.

- Every *N* time steps rescale the weights in each cell so that the peak weight is 1. (*N* = 5)

2.10 Problems Multiple Object Tracking

2.10.1 Congestion

Condensation is effective for tracking a single object over a cluttered background as it can deal with temporal ambiguity. In the case of tracking multiple objects or players on a football pitch the principal problem is not temporal ambiguity as a result of the measurement model interpreting the background to be the target object. The main problem is when players come together. This produces congested images data. If these players have similar measurement models, that is they are on the same team, when the players come together particles start to wander between the players exhibiting the multi-modal nature of the condensation algorithm. Ultimately the result is that all the particles end up tracking the player with the higher posterior probability. This is a similar problem to sample impoverishment in that weak modes are discarded. The result is that two clusters of particles track the same object.


Figure 17

Figure 17 contains two players. The particle distribution of the left player is shown in red and the right player in blue. Red pixels are distributed over each of the players. This exhibits the multi modal distribution, a result of the measurement model for the left player locating accurate matches at both players positions in the observed data. This is common when players on the same team come together. Their object representations are similar as they are wearing the same colour football kit, further from this the measurement model is designed to allow a degree of change in the representations. The result is that it is not possible to rely on individual object representation. In order to track multiple objects it is necessary to control the distribution of particles.

2.10.2 Occlusion

Occlusion occurs regularly in football matches. Partial occlusion produces a situation where the candidate regions can only represents a proportion of the target object thus the similarity between the target object representation and the candidate regions will be weak. Total occlusion produces a situation where there is no candidate regions which represents the target object.

The effect of occlusion is that particles will move to areas with higher posterior probability. As previously stated players on the same team have weak distinguishable features. The result is that multiple clusters will track the object with the higher posterior probability.

Issues of congestion and occlusion and be addressed by using contextual knowledge to re-weight the samples.

2.11 Re-weighting samples

The problem of congestion and occlusion is that particles wander to the highest posterior probability regardless of contextual information such as the number of objects being tracked or the location of objects. A method is used here which re weights samples based on this contextual information.
A closed world approach as defined in [65] is used. By using high level knowledge it is therefore possible to state that there are *N* objects being tracked at time t and the objects initial positions are known.

Local closed worlds are defined using proximity detection. The Euclidian distance between the known positions of the objects is computed in the image coordinate system. If these objects are close together they are said to exist in the same closed world. Otherwise players exist in individual closed worlds.

Contextual knowledge is used to identify each object within the closed world.


Figure 18

Figure 18 shows a closed world. It is known that the closed world contains two objects. It is possible to use this knowledge to prevent the affects of congestion and occlusion. In figure 18 two clusters are visible, the red cluster and the blue cluster. The size of these clusters are evaluated by computing the average Euclidian distance of particles from the cluster centre (taking the weighted mean as the centre of the cluster). The cluster with the larger average distance is termed the *weak cluster* this is because the cluster is divided into two sub-clusters. In figure 18 the *weak cluster* is shown in red and the *strong cluster* is shown in blue. The *weak cluster* consists of two sub-clusters, the *dominant* cluster which is on the left and the s*econdary* cluster which is on the right and is the results of candidate regions another object production higher posterior probability.

In order to maintain tracking of two objects the *secondary* cluster of the *weak* cluster needs to be adjusted to reflect that it is tracking an already tracked object, in addition the dominant cluster needs to be reinforced. This is done by re-weighting the samples in the *weak* cluster according to their location. The bounding box used in the objects' representation is taken as the area of observed data which represents an object. The weak cluster's particle which are located within this bounding box of the strong cluster are diminished to reflect that this candidate region represents an object in the closed world. Particles outside of the bounding box are reinforced as they represent the second object in the closed world.


Figure 19

Through experimentation it was found that dividing the *secondary* cluster by 20 and multiplying the *dominant* cluster by 10 produced the best results.

This approach works well for occlusion and scales well to close worlds with numerous objects as demonstrated by the footage shown in figure 20

Figure 20

When dealing with more than two objects in a closed world objects are processed in pairs. In figure 20 this refers to the blue and green clusters are re-weighted, the blue and pink clusters are re-weighted finally the pink and green clusters are re-weighted. This means that as congestion increases so does the size of the re-weighting.

This approach is capable of dealing with temporal occlusion. In the idealized case of two players A and B where p and q are their weights respectively. As A moves in front of B, B becomes partially occluded. This results in q diminishing. In addition to this particles from the cluster associated to B will spread to the higher posterior density of A forming sub clusters in B's particles. Therefore the occluded object becomes the *weak* cluster. If objects become totally occluded the particles associated with B spread to try and find a good candidate region. Once the occlusion is over the particles once again the candidate region can be identified

2.12 Tracking the Football

Figure 21

The footballs' representation in the image plane can be seen in figure 21. It is outlined by the red box. During the sequence of images the ball is represented by a circular area in the image plane with a radius of about 5 pixels. This area is significantly less than the area of the players.
Although the ball's shape does not deform in the same manner as the players, it is subject to occlusion more regularly and for longer periods of time. It also requires a prediction model which can cope with greater velocities and more sudden changes in direction. For these reasons no automated ball tracking has been employed. Instead the ball is tracked manually using a point and click graphical user interface.

2.13 Determining Players' Location in the Image Coordinate System

The condensation algorithm produces a multi-modal distribution.

For each player the condensation algorithm produces a distribution of probabilities (weights) typical to the graph shown in 20.

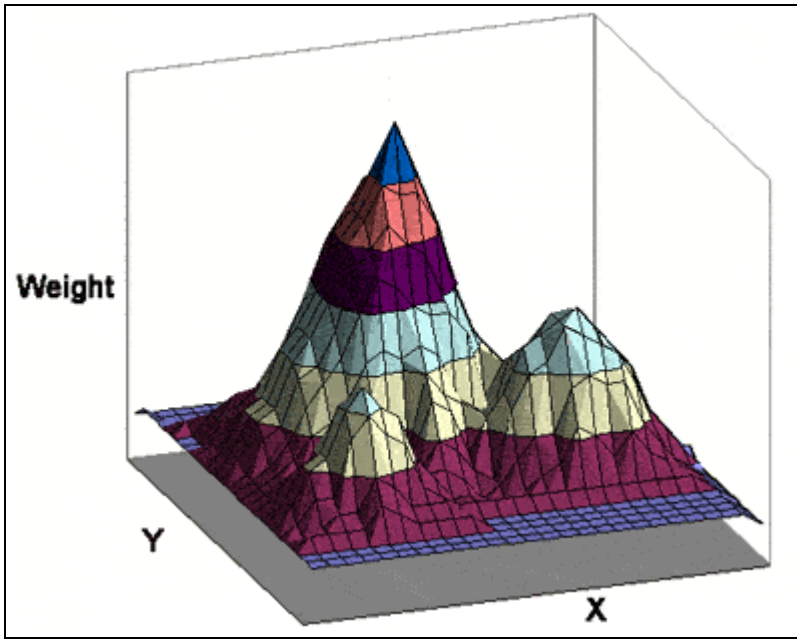


Figure 22

The example distribution has three peaks or modes. It is necessary to choose one of these weights to represent the location of the player in the x,y image coordinates subsequently allowing the calculation of the players real world coordinates. It is intuitive to simply select the position of the highest weight. The problem with this approach is that over a short period of time the dominant peak in 20 will remain in approximately the same place; however, the position of the maximum weight may alter within the area of the dominant peak.

This results in the players position jumping about. This is overcome by taking a weighted mean of the top 100 weights.

$$x = \frac{1}{\sum w_i} \left(\sum x_i * w_i\right) \qquad y = \frac{1}{\sum w_i} \left(\sum y_i * y_i\right)$$

By selecting only the top 100 weights, only one mode or peak is selected therefore avoiding the issue of calculating the weighted mean of multiple modes.

2.14 Recovering the Players' Location in the World Coordinate System

In order to recover real world coordinates from an image it is necessary to calibrate the camera. Camera calibration is the operation of estimating the camera's parameters. There are two types of parameters:

Intrinsic parameters - these describe how the camera forms an image

Extrinsic parameters - these describe the camera's position and orientation in the world coordinate frame.

2.14.1 Camera Model

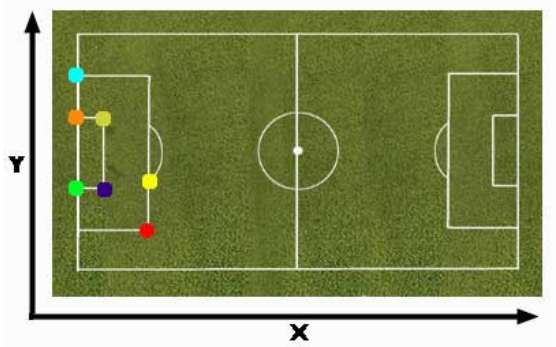
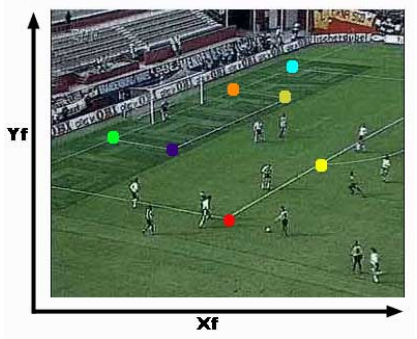
In order to recover these parameters a calibration system based on Tsai [81][82] [83] has been employed. This uses a camera model based on the pin hole model of perspective projection with 11 parameters.
Intrinsic parameters (5)

- **f** - effective focal length of the pin hole camera,
- **kappa1** - 1st order radial lens distortion coefficient,
- **Cx**, **Cy** - coordinates of centre of radial lens distortion and the piercing point of the camera coordinate frame's Z axis with the camera's sensor plane,
- $\mathbf{s_x}$ - scale factor to account for any uncertainty due to framegrabber horizontal scanline resampling,

Extrinsic parameters (6)

- **Rx**, **Ry**, **Rz** – represent the three rotation angles for the transform between the world and camera coordinate frames.
- **Tx**, **Ty**, **Tz** - represent the three translational components for the transform between the world and camera coordinate frames.

The calibration data used by this model consists of 3D (x,y,z) world coordinates of a point and corresponding 2D coordinates (Xf,Yf) (in pixels) of the point in the image. For this application coplanar calibration is used. This constrains the calibration points which lie in a single plane in the 3D world co-ordinate system.

Figure 23	Figure 24
	
3D (x,y,0) world coordinates of a point	Corresponding 2D coordinates (Xf,Yf)

Tsai 's method for camera calibration recovers the intrinsic orientation and the extrinsic orientation as well as the coefficients of the power series which models distortion.
This information isused to calibrate the camera. Once the camera has been calibrated it is possible to calculate world coordinates given image coordinates and image coordinates given world coordinates Details of the camera calibration method are given in appendix A.

2.15 Camera Motion

The footage used in this project has been chosen to minimises camera motion. There is however slight rotation and zooming. This motion does not affect the players relative positions but it does create slight displacement in the players world coordinate positions resulting in coordinates drifting in the 2D animation. This displacement is estimated and world coordinate positions are realigned accordingly.
The camera motion causes the background to appear as if it is moving. The result of this is that the stand and crowd enter the tracking area. This causes background clutter when tracking the goalkeeper.

3 Results

The tracking system has been successfully used to track 14 players over a sequence of images lasting around 4 seconds. Multiple objects are tracked over cluttered backgrounds (frames 70 to 80) in congested areas with occlusion (frames 50 to 70) and weak distinguishable feature representation.

Figure 25


Frame 0


Frame 10


Frame 20


Frame 30


Frame 40


Frame 50


Frame 60


Frame 70


Frame 80


Frame 90


Frame 100

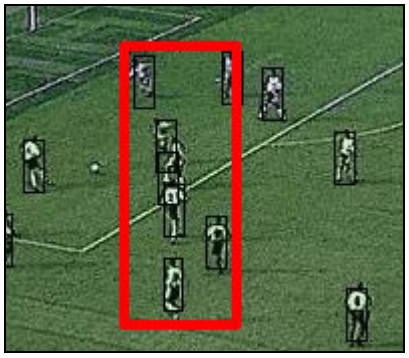


Figure 26

The tracker performs well on the selected footage showing that human motion can be modelled simply by using constant velocity and a constrained stochastic factor. The performance of the tracker was compared for a prediction model using the image coordinate system and a prediction model using the world coordinate system. The world coordinate system was found to be far superior producing much. Figure 26 shows frame 34 of the image sequence. There are several players lined up in close proximity. Whilst these players are close in the image coordinate system they are significantly spaced out in the world coordinate System. As a result the prediction model which uses the world coordinate system does performs better.

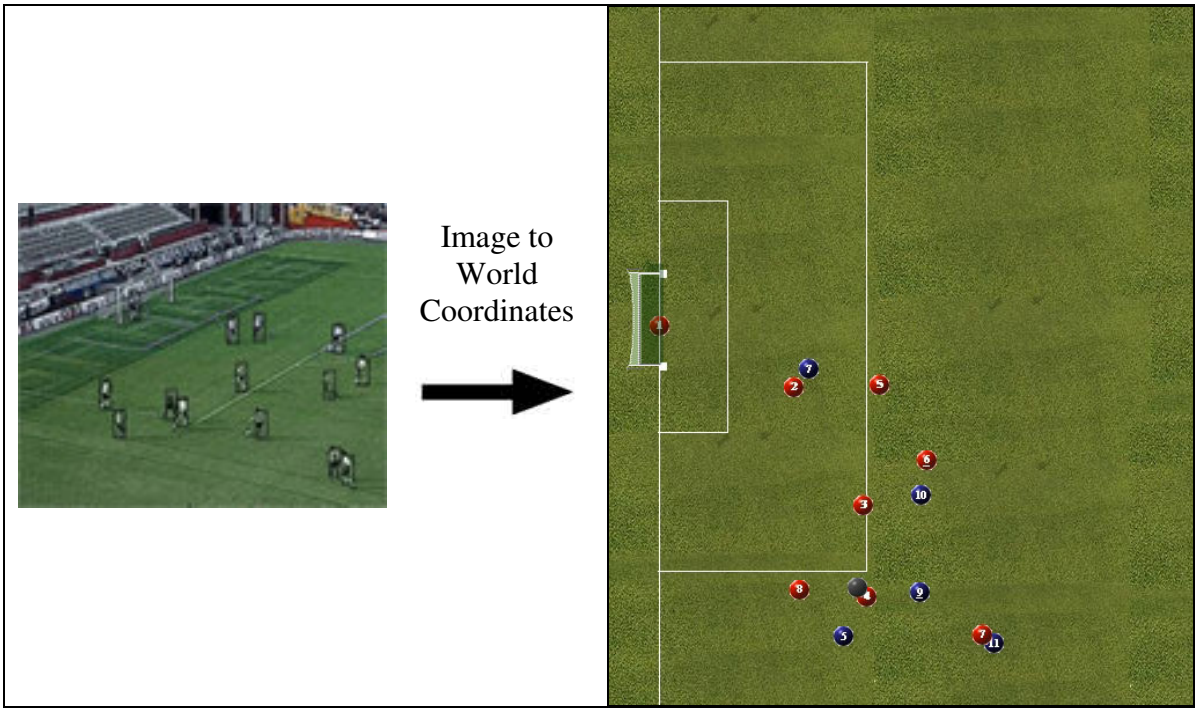


Figure 27

The location of these tracked objects is calculated in the world coordinate system. Figure 27 shows the positions of players at frame 0. The camera calibration system has an error of 0.1 yards when converting image coordinates to world coordinates. This is taken as an acceptable error; however, the motivation of this project is to recreate an animation of the goal. The primary objective of this animation is to convey the action of the goal not measure the accuracy of the calibration system.
This system has produced an animated goal for this footage. It is available at http://www.gameplanuk.cjb.net .

4 Design and Implementation

The aim of this project is to show a proof of concept however many of the design decisions have been made with future development of the product in mind.
This project has produced two pieces of software.

1) Tracking software
2) Match Consol – allows the user to watch an animated highlight.

4.1 Tracking software

The function of this software is to run the tracking algorithm on a sequence of images and, in addition to this provide useful functionality to help develop a tracking algorithm and analyse its performance.

The tracking software can be divided into the following components

- Graphical User Interface GUI I/O (Java)
- Low level pre condensation image processing (Java)
- Condensation Algorithm (C++)
- Camera Calibration (C)

An overview of the relationship between these components is shown in figure 28.

These components have been developed separately and communicate through a variety of interfaces. It is therefore possible to replace any of these components, for example camera calibration, in future work with minimum disruption to the Software.

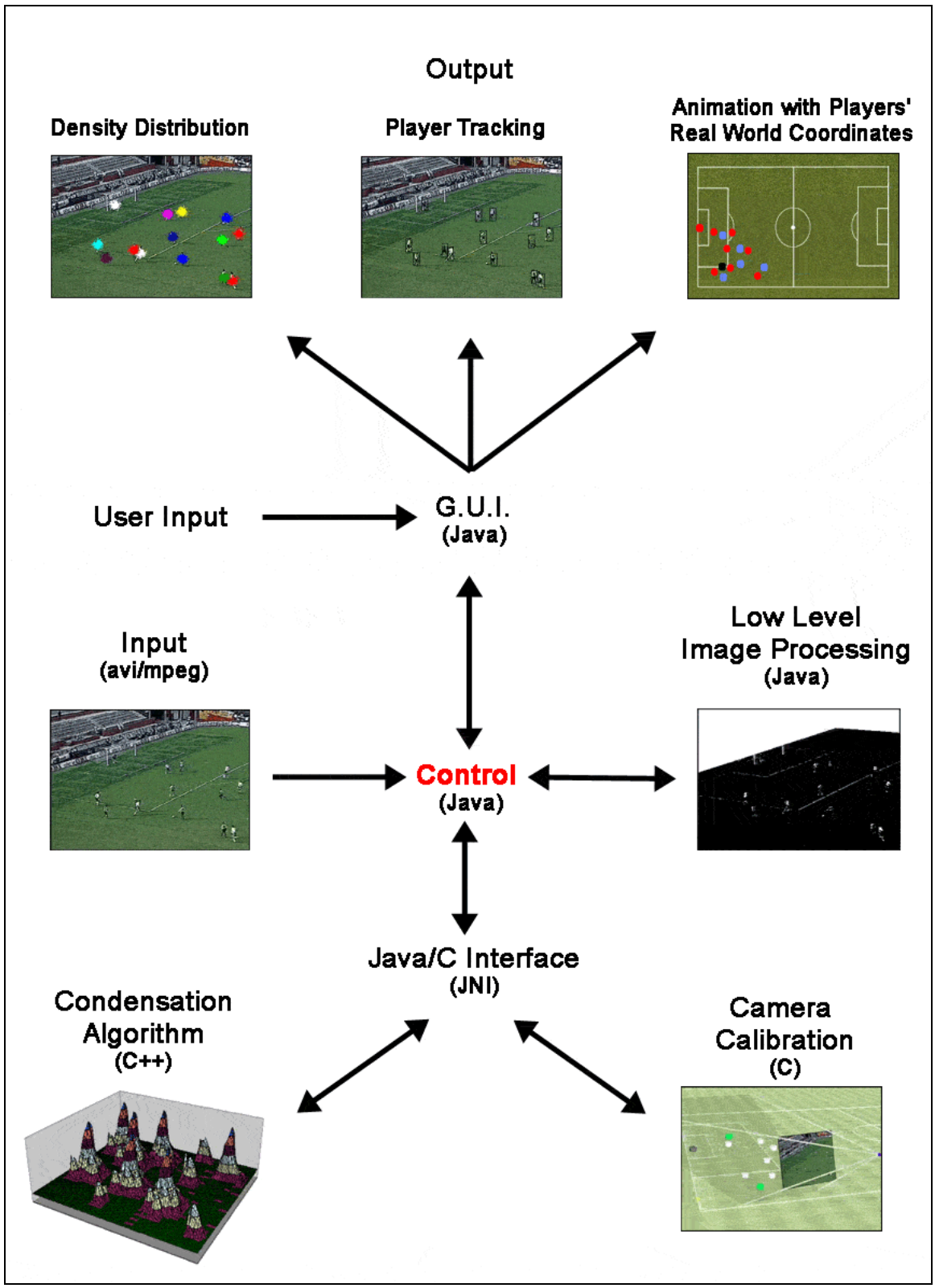

Output
Density Distribution
Player Tracking
Animation with Players' Real World Coordinates
User Input
G.U.I.
(Java)
Input
(avi/mpeg)
Low Level
Image Processing
(Java)
Control
(Java)
Java/C Interface
(JNI)
Condensation
Algorithm
(C++)
Camera
Calibration
(C)

Figure 28

4.1.1 Control

Control is the central program in this piece of software. It contains the main run method and initialises all the other parts of the program.
Within Control are the objects which represent each player. These objects contain world and image coordinates for each player for each image in the input sequence along with their target object representation, their corresponding graphics for animation and several other variables.

4.1.2 Graphical User Interface GUI

The aim of the GUI was to create a debugging environment for the tracking algorithm. The performance of the CONDENSATION algorithm is sensitive to a number of variables which control the prediction, measurement and background models. The GUI created an environment in which the tracking configuration could be easily changed and the affects of changes analyzed .
The GUI is written in Java because it was seen as the most appropriate language for dealing with video input, user input and image output. Methods are supplied by Java for creating and writing images. The majority of user input is via the mouse and occurs when initialising the players' positions and tracking the ball. Sun provide a package for dealing with audio and video known as the Java Media Framework JMF (2.11). This allows video footage of a football match to be streamed from a server to an application. JMF also provides the frame grabbing technology which is capable of extract individual frames for processing.

The GUI has been developed to provide a tool which aids in the analysis of the tracking algorithm. An interface similar to those used in image editing suites such as Adobe Photoshop and G.I.M.P has been implemented. This provides functionality such as painting over regions to remove them from the tracking area as well as displaying the image coordinates of the mouse position along with the RGB colour values and the HSB values. The current frame number is displayed and the current frame can be changed by entering the desired frame in the editable text box and selecting the GoTo button.

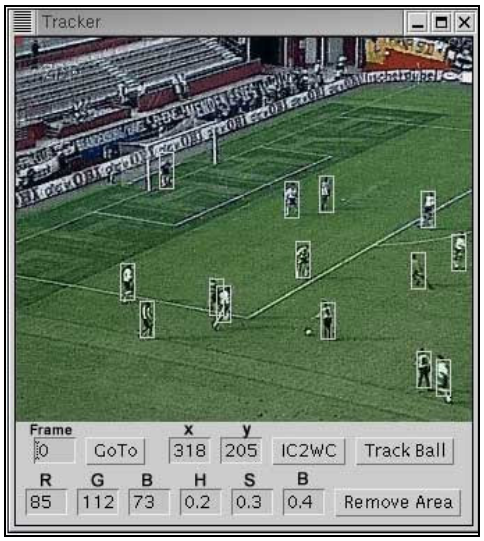


Figure 29

In order to make analysis easier it is possible to save the output. This allows the results of the tracking to be viewed after the program has been shut down. In addition to this the GUI can load saved data. The implication of this is that it is possible to make changes in the tracking algorithm and them compare these changes with previously saved outputs.

As the condensation algorithm is sensitive to minor changes in parameters, the ability to compare different tracking configurations was found to be a valuable tool in analysing the effects of changes to the configuration.

Time control

Video footage of a football match is a time based media. Time is controlled in two ways. There is a central control panel which has play, stop, skip frame forward, skip frame back and reset. This controls all windows opened by the program. Consequently allowing the comparison of output from different tracking configurations,

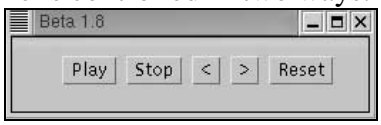


Figure 30

the times of individual sequences of images can be controlled by the individual windows.

4.1.3 Input

The input video format used was the Cinepak Codec by Radius with minimum compression and all the frames taken as key frames. This provided a good rate of compression without loss of information.

At the initial stages of development Matlab was used to extract images from video files. This process required an additional C program to convert these images into a usable format. This was a time consuming process. For that reason it was replaced with a Java program which could read video files.

The additional benefits of using video files are: it is the format the data is acquired in and it also allowed for simulations to be used instead of real footage throughout development. During the early stages of developing the tracking algorithm, simulations were used which were built using Ulead3D. These simulations represented simple objects moving around in 2D and 3D space. They were used to test and develop the tracking algorithm under a constrained environment.

4.1.4 Output

The Density Distribution and the Player tracking outputs are in the form of sequences of JPEG images. Sequences of images were chosen instead of video files for development and analysis purposes. Watching video requires a viewer with the appropriate Codec installed on the other hand a sequence of images is more portable. It can be viewed on Linux and Unix using XV or on Windows using Adobe Image Ready. If one of these programs is not available it is also possible to view individual frames using generic image viewers such as a web browser.

The Animation of the players real world coordinates is displayed using Java 2D graphics. The reason for this is future product development. The animation is designed to be downloaded over the internet. If this animation is in image or video format it is likely to be considerable in size taking longer to download. By using Java the animation can be viewed in a web browser as an Applet. This approach requires only a minimal number of graphics to be downloaded along with the positions of the graphics over a period of time.

4.2 Java™ Native Interface JNI 1.1

Java provides a standard programming interface for writing Java native methods called the Java™ Native Interface (JNI)
JNI allows code written in the C and C__ programming languages to be called from a Java program by declaring a native Java method, loading the library that contains the native code, and then calling the native method.
JNI is restricted in what it can pass and return between the languages. The main constraint is that it is only possible to pass primitive objects from C++ to Java. When initialising each time step of the condensation algorithm Java passes C++ an array of values which it should return an updated set of values; however, this is not possible. Several interfaces were built to facilitate the combination of C and Java.
Other methods of communicating between the languages were considered such as writing to shared files or the command line; however, these were considered more complicated and potentially harder to debug.

4.3 Low Level Image Processing (Pre CONDENSATION)

Pre CONDENSATION image processing such as background subtraction was implemented in Java in order to make the algorithm more efficient. Image processing tasks can be carried out while the images are being extracted from the video files.

4.4 CONDENSATION Algorithm

It is not only beyond the capacity of this project to implement this algorithm but it is also unnecessary as several implementations already exist. In this project an implementation by David Tweed is used. The implementation was developed specifically to tracking wildlife using Subordinate CONDENSATION; however, it provides a framework for the algorithm which can be used for this project.
The framework provided by Tweed's implementation did not cover all aspects of the algorithm. Initialisation, measurement and prediction models were implemented specific to the tracking of football players. These components were developed so that they could be plugged into the CONDENSATION algorithm. This provided a platform for experimenting with various models.
Tweeds implementation is written in C++ for Unix and Linux platforms.

4.5 Image Processing Library IPLIB

Handling images in C and C++ was done using the IPLIB image library[91]. This provided all the necessary functions to manage images with RGB space.

4.6 Camera Calibration

Calibration needed to be done with the absence of knowledge of the camera's parameters and with information from only one camera. Several methods of calibration were considered with a view to implementing the most appropriate. Methods investigated included a two-step camera calibration process based on linear least squares formulations [84] and Vanishing points [85].

It was decided that implementing a camera calibration algorithm was an unnecessary use of time as generic calibration software already exists which could be adapted. Camera calibration was achieved using Tsai Camera Calibration Library[90]. The Library provides routines for calibrating perspective projection camera models. The library is written in C and a C++ interface written by Chris Needham[92] has been used.
The Tsai calibration library is limited in the degree of rotation between world coordinates and image coordinates for which it can correctly calibrate the camera. The world coordinates are rotated about a point prior to the calibrations.

4.7 Match Consol.

The match consol is a Java program which displays 2D animations of goals and highlights. Java 2D graphics are used with a view to downloading these animations over the internet. Figure 31 shows the match consol.

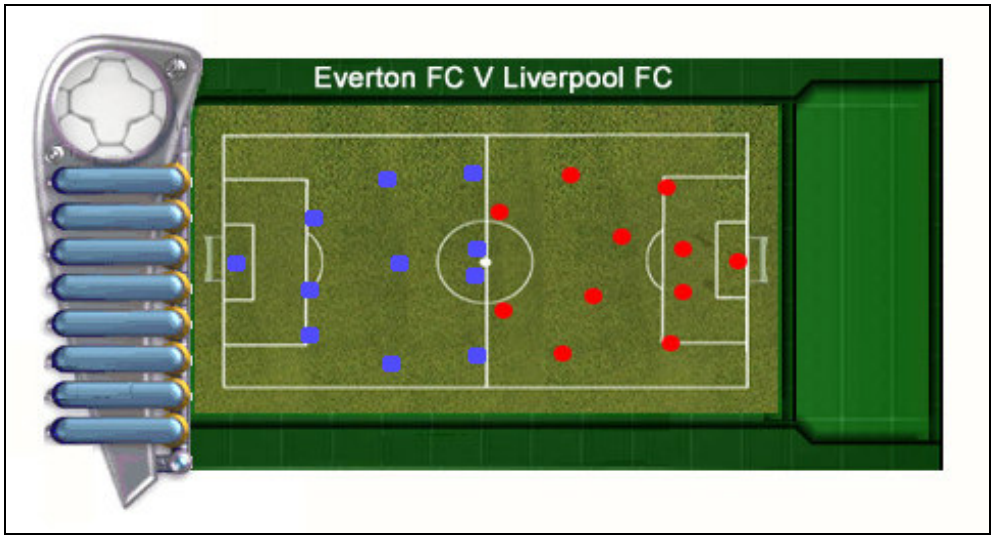


Figure 31

4.8 Bugs

The reconstruction process requires a significant amount of memory to run. The size of the default memory allocated to the Java virtual machine is not sufficient. This needs to be increased to 1024 Mbytes in order to run without aborting.
The process also requires free space to write output. If insufficient space is not available the program can not run.

4.9 Obtaining Footage of Football Matches

For the purposes of this project footage was required of a football match from a relatively static camera with high elevation. Attempts were made to obtain footage from static camera by contacting the BBC and Sky Sports. Several tapes were provided by Steve Whitehead of the BBC sports archive however the footage generally contained significant panning and zooming. An attempt was also made at obtaining footage by filming a football match between Bristol City and Peterbrough United. Press passes and access to the gantry were arranged by Ed Furniss of the Bristol City Press Office. Unfortunately a combination of a weak tripod and rickety gantry resulted in significant camera motion. Figure 32 shows a photo taken during filming. The footage used in this project is taken from Bebie [65]


Figure 32

5 Current state

The condensation algorithm has been used to track multiple objects through congested areas. A method of occlusion reasoning has been implementation using a closed world assumption and high level contextual knowledge.

The results of this project show that it is possible to track players using the approach described in this thesis.
Serious issues in tracking have been successfully addressed using real world data.
Multiple players have been tracked through occlusion, congestion, background clutter.

This project has successfully implemented a reconstruction system which generates a 2D animated goal from short video sequences of a football match. This demonstrates that the concept of tracking multiple players and recover their world coordinates is achievable.

This project has achieved a proof of concept. From a business view this means that the product is in a situation where it can be taken to potential investors with aim to obtain financing for further development.

6 Future Work

This work set out to achieve a proof of concept. The work has therefore concentrated on proving this concept on a single piece of footage. Further work will involve developing a more general a models to increase the robustness of the system.

6.1 Automatic object initialisation

The current implementation requires manual initialisation. The calibration points are initialised by clicking on known points in the image plane and the objects which are tracked are initialised by dragging a box around the blob representing them. By building a generic object model offline prior to the tracking, it should be possible to implement an algorithm to automatically detect the presence of objects. This will decrease the manual intervention in the process but more importantly it will allow new players entering the image plane to be recognised. Currently the only objects which are tracked are those which exist, at and are initialised in, the first frame of the sequence of images. Objects entering the image plane are not tracked. In practice it is often acceptable to track only the objects present at the start; however, the presence of an untracked object affects the occlusion resolution. A tracked player occluded by a non tracked player may cause the tracker to switch to the untracked player if it is a better fit.
A possible approach to this problem may be to use bootstrapping of the models using generic models from a database and deforming them to suite the specific target objects.

6.2 Action Recognition

The current implementation assumes the players are in contact with the ground plane at all times. This assumption is violated when players jump to head the ball. Action and gesture recognition is a prominent area of research and a great deal of work has been done [86] There is scope to extend this work to analyse the motion of football player in an attempt to recognise action such as jumping and passing of the ball. Such a system would allow for players jumping and has the capacity to produce an automated football analysis system. However, implementing a system to perform action recognition is a significant amount of work and for the purposes of recovering players' real world positions it is unnecessary.

The assumption that players are constantly in contact with the ground plane is taken to be an acceptable assumption.

6.3 Ball Tracking and Improved Camera Calibration

The current implementation assumes the football is in contact with the ground plane at all times. This assumption is regularly violated. To recover the real world coordinates of the football, a calibration technique which employs a non-coplanar model allowing the calibration points to occupy a 3D volume is required. This is achievable with the Tsai system.
In [87] Reid and North propose a system for extracting the 3D trajectory of a football using shadows. The technique is robust to camera motion, rotation and zoom but not translation. Use is made of shadows on the ground plane to calculate the vertical projection of the ball onto the ground plane in order to compute the balls height above the ground. The main hindrance of this technique is the footage to which it can be applied. It is constrained to footage of football matches in which clear shadows can be identified. Modern stadiums can cause sharp shadows on the pitch, blocking out sunlight and therefore preventing the ball from casting a shadow. However this technique can still be applied to matches played under floodlights. Floodlights often create multiple shadows which provide additional information for tracking.

6.4 Improved Prediction Model

It is possible to improve the deterministic element of the prediction model by using contextual information. This would involve developing individual prediction models which reacted to the players surroundings. This is particularly relevant to the tracking of goal keepers whose motion is predominantly reactionary but it is also applicable to outfield players as their movements are a response the their surroundings.

6.5 Locally Discriminating Adaptive Background Model

The implementation of a static background model has worked well on the presented data. However a static background is not suitable for longer periods of time as the shadows will alter the background. In addition to this camera motion will alter the background. To overcome there problems an adaptive background model for the entire pitch can be generated which is capable of discriminating between background and foreground objects using local contextual information.

6.6 Automatic Camera Recalibration

An automatic camera recalibration system has been designed and implemented which uses a Hough transform to detect lines in the image plane. These lines represent pitch markings and can be used to recover the location points in the image plane which have known

corresponding points in the world coordinate system. This information can be used to recalibrate the camera if these features have moved significantly as a result of panning and zooming. Unfortunately this system has not been successfully implemented as a result of lack of time.

6.7 Performance

The algorithm has been tested on machines with Athlon XP 1800+ processors and 512Mb RAM running Linux. Currently the reconstruction system does not run in real time however no optimisation have been attempted. Future work will include optimising the performance of the algorithm, including reducing the number of cross language function calls which is known to reduce the performance.

6.8 Future Applications

The motivation for this work was to create animations of football highlights. The is scope for development of this idea to other sports such as hockey, athletics, rugby, horse racing and motor sports. All of these application present unique and challenging problems which this work provides a framework for solving.

Tracking of sports players is also of interest to sports science industry. An automated player tracker can be used to recover information about a match which would otherwise be difficult to obtain. This information includes how much ground the player covers, the average speed and acceleration of players. Information such as this can be used for a number of application from designing specific training routines based, to testing the affects of a performance enhancing drink or food.

The information recovered form the tracking of sports players can be used to analyse how the sport is played. This is particularly interesting in sports which are complicated and require a large amount of interactivity. This information can be used to analyses how players interact with each other, how they make use of space and how they react under certain circumstances.

7 References

[1] MODEL BASED TRACKING OF ARTICULATED OBJECTS Kevin Michael Nickels, Ph.D. Department of Electrical and Computer Engineering University of Illinois at Urbana-Champaign, 1998

[2] Multi-Scale Feature Tracking and Motion Estimation Lars Bretzner unpublished

[3] Interpreting Human Gesture with Computer Vision - Coutaz, Crowley (1995)

[4] J.A. Webb and J.K. Aggarwal. Visually interpreting the motion of objects in space. Computer, 14:40-46,1981.

[5] X.Feng and P.Perona,"Real Time Motion Detection System and Scene Segmentation", CDS Technical Report CDS 98-004, California Institute of Technology

[6] I.E. Fordon . Theories of Visual Perception, chapter 3, pages 46-75, John Whiley & Sons, New York, 1989

[7] p1 Active Contours by A.Blake and M. Isard Springer press

[8] Vision for Longitudinal Vehicle Control Philip F. McLauchlan and Jitendra Malik, EECS Computer Science Division, University of California at Berkeley, Berkeley

[9] C Setchell. Applications of Computer Vision to Road-traffic Monitoring. September 1997. unpublished

[10] Chris G Perrott and Leonard G C Hamey. Object recognition, a survey of the literature. Technical Report 91-0065C, School of MPCE, Macquarie University, NSW 2109 Australia, January 1991

[11] D. Koller, K. Daniilidis, T. Thorhallson, and H.-H. Nagel. *Model-based object tracking in traffic scenes*. Proc. ECCV, pp. 437--452. Springer-Verlag, Santa Marguerita, 1992

[12] University of Reading (Baker and Sullivan, 1992; Sullivan, 1992).

[13] J.O'Rourke and Badler "Model-based image analysis of human motion using constraint propagation" (1980) PAMI 2(6)

[14] D. Hogg, Model-based Vision: A Program to See aWalking Person, Image and Vision Computing, 1(1), pp.5-20, 1983.

[15] Model-based Tracking of Human Walking in Monocular Image Sequences Huazhong Ning, Liang Wang, Weiming Hu and Tieniu Tan National Laboratory of Pattern

Recognition Institute of Automation, Chinese Academy of Sciences, Beijing, P. R. China, 100080

[16] T.-J. Cham and J.M. Rehg. A multiple hypothesis approach to figure tracking. In *CVP*R, June 1999.

[17]H. Sidenbladh, M. Black, and D. Fleet. Stochastic tracking of 3d human figures using 2d image motion. *ECC*V, pp. 702–718, 2000.

[18]S. Wachter and H. Nagel. Tracking of persons in monocular image sequences. *CVI*U,74(3):174–192, 1999.

[19] Automatic detection and tracking of human motion with a view-based representation Ronan Fablet and Michael J. Black EUR. CONF. ON COMPUTER VISION, ECCV'02, MAY 2002, COPENHAGUEN

[20] Bülthoff, I., H.H. Bülthoff and P. Sinha: View-based representations for dynamic 3D object recognition. Max Planck Institute for Biological Cybernetics (feb 1997)

[21] Turk, M. and Pentland, A. (1991). Face recognition using eigenfaces, *Proc.IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognitio*n, Maui, pp. 586–591

[22] ui, Y. and Weng, J. J. (1996). View-based hand segmentation and hand-sequencerecognition with complex backgrounds, International Conference on Pattern Recognition

[23] Black, M. J. and Jepson, A. D. (1998b). Eigentracking: Robust matching and tracking of articulated objects using a view-based representation, Int. J. of Computer Vision 26(1): 63–84.

[24] A. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," CVPR, 1994

[25] B. W. Mel, "SEEMORE: Combining color, shape, and texture histogramming in a neurally inspired approach to visual object recognition," Neural Computation, vol. 9, no. 4, pp. 777--804, 1997

[26] Koller, D, Weber, J, Malik, J (1994a) "Robust multiple car tracking with occlusion reasoning", *ECC*V, Stockholm, Sweden, pp 189-196.

[27] Koller, D, Weber, J, Huang, T, Malik, J, Ogasawara, G, Rao, B, Russell, S (1994b) "Towards robust automatic traffic scene analysis in real-time.", *ICP*R, Israel, Vol 1, pp 126-131.]

[28] A. M. Baumberg and D. C. Hogg, An efficient method for contour tracking using active shape models, in *Workshop on Motion of Non-Rigid and Articulated Objects, Austin, TX, 199*4, pp. 2–14

[29] G. Rigoll, S. Eickeler, and S. M uller, Person tracking in real world scenarios using statically methods, in *The Fourth International Conference on Automatic Face and Gesture Recognition, Grenoble, France, March 200*0.

[30] S. Huwer & H. Hiemann, “2D Object Tracking Based on Projection Histograms”, FORWISS Knowledge Processing Research Group, Bavarian Research Center for Knowledge Based Systems, Am Weichselgarten 7, D-91058 Erlangen. Denmark

[31] Pfinder: Real-Time Tracking of the Human Body 1997 Christopher Wren, Ali Azarbayejani, Trevor Darrell, Alex Pentland IEEE Transactions on Pattern Analysis and Machine Intelligence

[32] Alex Pentland. Classification by clustering. In *Proceedings of the Symposium on Machine Processing of Remotely Sensed Data*. IEEE, IEEE Computer Society Press, June 1976

[33] R. J. Kauth, A. P. Pentland, and G. S. Thomas. Blob: An unsupervised clustering approach to spatial preprocessing of mss imagery. In *11th Int'l Symposium on Remote Sensing of the Environment*, Ann Arbor, MI, April 1977

[34] C. Ridder and O. Munkelt and H. Kirchner daptive background estimation and foreground detection using Kalman filtering In Proc. ICAM, 193--199, 1995

[35] S. J. McKenna, Y. Raja, and S. Gong. Tracking colour objects using adaptive mixture models. *Image and Vision Com-puting*, 17(3-4):225–231, 1999.

[36] C. Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *CVP*R, pages 246– 252, Fort Collins, Colorado, June 1999

[37] A Real-Time Computer Vision System for Vehicle Tracking and Traffic Surveillance Benjamin Coifman unpublished Institute of Transportation Studies University of California Berkeley, California, 94720

[38] S. J. McKenna, S. Jabri, Z. Duric, and H. Wechsler, "*Tracking interacting people*," in Proc. Int. Conf. Automatic Face and Gesture Recognition, Grenoble, France, 2000, pp. 348—353

[39] T. B. Moeslund and E. Granum, Multiple cues used in model-based human motion capture, in *TheFourth International Conference on Automatic Face and Gesture Recognition, Grenoble, France, March 200*0.

[40] K. Rohr, *Human Movement Analysis Based on Explicit Motion Model*s, chap. 8, pp. 171–198, Kluwer Academic, Dordrecht/Boston, 1997

[41] R. Okada, S. Yamamoto & Y. Mae, “Realtime Person Tracking System”, Department of Mech. Eng. for Computer-Controlled Machinery, Osaka University, 1995

[42] Y. Shirai, T. Yamane & R. Okada, "Robust Visual Tracking by Integrating Various Cues", IEICE TRANS. INF. SYST., Vol. E81-D, No. 9 September 1998, pp. 951-958

[43] Robust Feature Tracking Ben Galvin, Brendan McCane and Kevin Novins. *Submitted to Fifth International Conference on Digital Image Computing, Techniques, and Applications (DICTA), December 6-8, 1999, Perth, Australia*

[44] S.M. Smith Real-Time Motion Segmentation and Object Tracking Technical Report TR95SMS2b (Shorter versions of this have now been published in PAMI and ICCV95)]

[45] J.A. Webb and J.K. Aggarwal. Visually interpreting the motion of objects in space. Computer, 14:40{46,1981}

[46] Visual Tracking Ying Wu Electrical & Computer EngineeringNorthwestern University Evanston, IL 60208 ECE510-Computer Vision Notes Series 7

[47] The condensation algorithm: A literature Survey, Jeff Brasket February 2002

[48] ECSE 6650 -: Features Tracking and Shape & Structure from Motion Peng-Jui Ku, Lu-Yun Chen, Jie Zou December 3, 2001

[49] Kalman,R.E.1960."A New Approach to Linear Filtering and Prediction Problems," Transaction of the ASME —Journal of Basic Engineering, pp.35-45 (March 1960).

[50] INTRODUCTION TO KALMAN FILTERS E V Stansfield Thales Research Ltd, Reading Kalman filter tutorial

[51] ECSE 6650 - Computer Vision Project #4: Features Tracking and Shape & Structure from Motion Peng-Jui Ku, Lu-Yun Chen, Jie Zou December 3, 2001

[52] A. M. Baumberg. *Learning Deformable Models for Tracking Human Motio*n. PhD thesis, School of Computer Studies, University of Leeds, 1995.

[53] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligenc*e, 19(7):780–785, 1997.

[54] Tracking Cars in Range Images Using the Condensation Algorithm Esther B. Meier and Frank Ade Communication Technology Lab, Image Science Swiss Federal Institute of Technology (ETH) CH-8092 Zurich, Switzerland

[55] The CONDENSATION Algorithm: A Literature Survey Jeff Brasket February 2002 Unpublished

[56] Visual Tracking Ying WuElectrical & Computer Engineering Northwestern UniversityEvanston, IL 60208 ECE510-Computer Vision Notes Series 7

[57] Tracking Cars in Range Images Using the Condensation Algorithm Esther B. Meier and Frank Ade Communication Technology Lab, Image Science Swiss Federal Institute of Technology (ETH) CH-8092 Zurich, Switzerland

[58] Tracking Cars in Range Images Using the Condensation Algorithm Esther B. Meier and Frank Ade Communication Technology Lab, Image Science Swiss Federal Institute of Technology (ETH) CH-8092 Zurich, Switzerland

[59] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *European Conference on Computer Visio*n, volume 1, pages 343–356, 1996.

[60] M. Isard and A. Blake. Condensation conditional density propagation for visual tracking. *International Journal on Computer Visio*n, 29(1):5–28, 1998.

[61] Tracking Cars in Range Images Using the Condensation Algorithm Esther B. Meier and Frank Ade Communication Technology Lab, Image Science Swiss Federal Institute of Technology (ETH) CH-8092 Zurich, Switzerland

[62] Tracking Many Objects Using Subordinated CONDENSATION David Tweed and Andrew Calway. In Paul Rosin and David Marshall, editors, *Proceedings of the British Machine Vision Conference*, pages 283--292. BMVA Press, October 2002.

[63] Computer Graphics Forum Volume 19, Issue 3 (August 2000) Practice and Experience:A Video-Based 3D-Reconstruction of Soccer Games T. Bebie, H. Bieri Department of Computer Science and Applied Mathematics, University of Berne, Switzerland

[64] S.S. Intille and J.W. Davis and A.F. Bobick, Real-Time Closed-World Tracking, Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society Press, June 1997, pp. 697-703.

[65] S.S. Intille and J.W. Davis and A.F. Bobick, Real-Time Closed-World Tracking, Massachusetts Institute of Technology, MIT Media Lab Perceptual Computing Group Technical Report No. 403, November, 1996.

[66] S.S. Intille and A.F. Bobick, Closed-World Tracking, Proceedings of the International Conference on Computer Vision (ICCV), June 1995, pp. 672-678. This official proceedings is missing one page of the article, so here's the full text.]

[67] S.S. Intille and A.F. Bobick, Visual Tracking Using Closed-Worlds, Massachusetts Institute of Technology, MIT Media Lab Perceptual Computing Group Technical Report No. 294, November, 1994. Complete version of ICCV paper.

[68] Orad Hi-Tec Systems. *VirtuaReplay*. http://www.orad.co.il

[69] Y. Seo, S. Choi, H. Kim and K.S. Hong, "Where are the ball and players? Soccer game analysis with color-based tracking and image mosaick," International Conference on Image Analysis and Processing, Sept. 1997, Florence, Italy

[70] Needham, C J; Boyle, R D. *Tracking multiple sports players through occlusion, congestion and scale* in: British Machine Vision Conference 2001, vol. 1, pp. 93-102 BMVA. 2001

[71] I. Kakadiaris and D. Metaxas, Vision-based animation of digital humans, in *Conference on Computer Ani-mation, 199*8, pp. 144–152

[72] Errors and Mistakes in Automated Player Tracking Janez Per s Marta Bon and Stanislav Kova ci c Faculty of Electrical Engineering, Faculty of Electrical Engineering, Bostjan Likar (Ed.): Proceedings of Sixth Computer Vision Winter Workshop, Bled, Slovenia, February 7-9, 2001, pp. 25-36

[73] Christopher R. Wren, Ali Azarbayejani, Trevor Darrell, Alex Pentland Pfinder: Real-Time Tracking of the Human Body (1995) IEEE Transactions on Pattern Analysis and Machine Intelligence, July 1997, vol 19, no 7, pp. 780-785

[74] Stephen J. McKenna, Sumer Jabri, Zoran Duric, Harry Wechsler. Tracking Interacting People. In Proc.Fourth IEEE int. Conf. Automatic Face and Gesture Recognition, pages 348-353,2000

[75] Chris Stauffer and W.E.L Grimson."Adaptive background mixture models for real-time tracking", *CVPR99,* Fort Colins, CO, (June 1999).]

[76]Color Gamut Transform Pairs, Proc. SIGGRAPH 78, 12-19,1978 A.Smith

[77] http://escience.anu.edu.au/lecture/cg/Color/HSV_HLS.en.html

[78] David Tweed and Andrew Calway. Tracking Many Objects Using Subordinated CONDENSATION. In Paul Rosin and David Marshall, editors, *Proceedings of the British Machine Vision Conference*, pages 283--292. BMVA Press, October 2002

[79] J MacCormick and M Isard. Partitioned sampling, articulated objects and interface quality hand tracking. In *ECC*V, pages 3–19, 2000.

[80] R Sedgewick. *Algorithm*s. Addison-Wesley, 1992.

[81] A versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses", Roger Y. Tsai, IEEE Journal of Robotics and Automation, Vol. RA-3, No. 4, August 1987, pages 323-344.

[82] "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision", Roger Y. Tsai, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, 1986, pages 364-374

[83] Horn, B. K. P., "Tsai's camera calibration method revisited", *unpublished*, 2000. Avaliavle at http://www.ai.mit.edu/projects/vision/publications/index.php 15/5/03

[84] Image-based Modeling Using a Two-step Camera Calibration Method,Paulo Cezar Pinto Carvalho, Flávio Szenberg, Marcelo Gattass, IMPA- Instituto de Matemática Pura e Aplicada

[85] 3D Reconstruction using three vanishing points from a single image, Y. Yoon, J. Im, D. Kim, J. Choi

[86] Stochastic tracking of 3D human figures using 2D image motion, Sidenbladh, H., Black, M. J., and Fleet, D.J., *European Conference on Computer Vision*, D. Vernon (Ed.), springer Verlag, LNCS 1843, Dublin, Ireland, pp. 702-718 June 2000.

[87] 3D Trajectories from a Single Viewpoint using Shadows I. D. Reid, A. North Department of Engineering Science Oxford University Oxford OX1 3PJ UK BMVC 98 - Viewing Real-World Imagery

[88] Ken Tabb and Neil Davey and R. G. Adams and S. J. George. Analysis of Human Motion Using Snakes and Neural Networks", AMDO, pages 48-57,2000

[89] Tracking and Analysis of a Sporting Event By: Cloretta Ceasar 801 Atlantic Drive Atlanta, GA 30332-0280 Georgia Institute of Technology Unpublished http://www.cc.gatech.edu/classes/AY2000/cs7495_fall/participants/clojo/fp/report.html (15/05/03)

[90] Reg WillsonNASA Jet Propulsion Laboratory California Institute of Technology, 4800 Oak Grove Drive, Mail Stop 125-209,asadena, California 91109-8099 http://www-2.cs.cmu.edu/~rgw/TsaiDesc.html 15/5/03

[91] http://www.cs.bris.ac.uk/%7Eandrew/index.html 15/5/03

[92] http://www.comp.leeds.ac.uk/chrisn/Tsai/index.html 15/5/03

8 Appendix A

8.1 Camera Calibration Techniques

The **Intrinsic Orientation** describes the relationship the between camera-centric coordinates and the image coordinates. The origin of the camera coordinate system is at the centre of the projection. The z axis runs along the optical axis while the x and y axis are parallel to the image x and y axis.
The Intrinsic Orientation can be defined by the perspective projection equation

$$\frac{x_I - x_0}{f} = \frac{x_C}{z_C} \qquad \frac{y_I - y_0}{f} = \frac{y_C}{z_C}$$

Where (x_I, y_I) are image coordinates and $(x_C\ y_C)$ are camera coordinates. f is the principle distance which represents the distance from the centre of projection to the image plane and (x_0,y_0) is the principle point. The centre of projection is at $(x_0,y_0,f)^T$ as measured in the image plane coordinate system

The **Extrinsic Orientation** describes the relationship between the scene centred coordinate system and a camera centred coordinate system. The transformation from scene coordinate system to camera coordinate system is made up of a rotation and a translation.
This transform can be represented as

$$r_c = R(r_s) + t$$

where r_s is the coordinates of a point in the scene coordinate system and r_c is the coordinates of a point measured in the camera coordinate system. t is the translation and R is the rotation.

Distortion occurs as a result of the spherical lenses used by optical systems. The effect is a geometric distortion in the radial direction. Displacement of points by this radial distortion are modelled using the equation

$$\delta x = x (k_1 r^2 + k_2 r^4 + \ldots\ldots\ldots)$$
$$\delta y = y (k_1 r^2 + k_2 r^4 + \ldots\ldots\ldots)$$

where x and y are measured form the centre of distortion (principle point) and r is the distance from the principle point.
k is the coefficients of the power series which will be recovered.

8.1.1 System used to recover camera parameters

The calibration data consists of a set of points in world coordinates and the corresponding points in the image. It is from this data that the camera's parameters are recovered. This is done in a two stage approach.

1) Estimate the maximum number of parameters possible using linear least squares fitting methods. This is done for convenience and speed and uses the pseudo-inverse matrix. Constraints between parameters are not enforced. In this initial step the error in the image plane is not minimised, instead a value which simplifies the analysis is minimised leading to linear equations. As these parameter estimations are only used as

starting points for the final estimation, using linear equations does not affect the final camera calibration.

2) In this step the remaining unknown parameters are obtained using a non-linear optimisation method. This method establishes the best fit between the observed image points and those predicted from the target model. In addition to this the parameters' estimations obtained in part 1 using linear equations are refined.

8.1.2 Recovering the rotation and translation matrix.

It is assumed that a reasonable estimate of the principle point (x_0,y_0) can be obtained. It is commonly in the centre of the optical sensor. Coordinates are referred to this point using

$$x'_I = x_I - x_0 \qquad \text{and} \qquad y'_I = y_I - y_0$$

such that

$$\frac{x'_I}{f} = s\frac{x_C}{z_C} \qquad \frac{y'_I}{f} = \frac{y_C}{z_C}$$

By taking a point in the image and considering only the direction as measured from the principle point it is possible to generate an equation which is independent of the unknown principle distance f and independent of the radial distortion.

$$\frac{x'_I}{y'_I} = s\frac{x_C}{y_C}$$

Using the coplanar assumption that $z_s = 0$ it is possible to expand this equation in terms of the components of the rotation matrix R.

$$\frac{x'_I}{y'_I} = s\,\frac{r_{11}x_S + r_{12}y_S + t_x}{r_{21}x_S + r_{22}y_S + t_y}$$

Through cross multiplication this produces a linear homogeneous equation with 6 unknowns r_{11}, r_{12}, r_{21}, r_{22}, t_x and t_y

$$(x_S\, y'_I)\,\mathbf{r_{11}} + (y_S\, y'_I)\,\mathbf{r_{12}} + (y'_I)\mathbf{t_x} - (x_S\, x'_I)\,\mathbf{r_{21}} - (y_S\, x'_I)\,\mathbf{r_{22}} - (x'_I)\mathbf{t_y} = 0$$

The coefficients of these unknowns are products of components of corresponding scene and image coordinates. Therefore this equation can be solved with 6 sets of corresponding scene and image coordinates. This produces a 2X2 rotation matrix which is then used to estimate the full 3X3 matrix.

This is achieved by estimating the scale factor of the rotation matrix

The rotation matrix is orthonormal

$$r'^2_{11} + r'^2_{12} + r'^2_{13} = k^2$$
$$r'^2_{21} + r'^2_{22} + r'^2_{23} = k^2$$
$$r'^2_{11}r'^2_{21} + r'^2_{12}r'^2_{22} + r'^2_{13}\, r'^2_{23} = 0$$

From the above 3 equations it is possible to obtain:

$$r'^2_{13} = k^2 - (r'^2_{11} + r'^2_{12})$$
$$r'^2_{23} = k^2 - (r'^2_{21} + r'^2_{22})$$

Where

$$k^2 = \tfrac{1}{2}\,[\,(r^2_{11} + r^2_{12} + r^2_{21} + r^2_{22}) + \sqrt{\{((r_{11} - r_{22})^2 + (r_{12} + r_{21})^2)((r_{11} + r_{22})^2 + (r_{11} - r_{21})^2))\}}\,]$$

The third row of the rotation matrix is calculated as a cross-product of the first two rows therefore completing the linear estimation of the rotation matrix R.

The first two components of the translation matrix t_x and t_y have been estimated. It is possible to estimate the remaining component t_z along with the principle distance f with the following equations:

$$\frac{x'_I}{f} = s\,\frac{r_{11}x_S + r_{12}y_S + r_{13}z_S + t_x}{r_{31}x_S + r_{32}y_S + r_{33}z_S + t_y}$$

$$\frac{y'_I}{f} = s\,\frac{r_{21}x_S + r_{22}y_S + r_{23}z_S + t_x}{r_{31}x_S + r_{32}y_S + r_{33}z_S + t_y}$$

As the rotation matrix is already estimated it is possible to use cross multiplication to formulate linear equations with two unknowns f and t_z. Corresponding scene and image coordinates can be used to solve these unknowns.

8.1.3 Non-Linear Optimisation

This optimisation minimises the image errors. Errors are measured as the difference between the actual observed image positions $(x_I,y_I)^T$ and the predicted positions $(x_P,y_P)^T$. The predicted positions are generated from the target coordinates $(x_S, y_S, z_S)^T$.

A modified Levenberg-Marquart method is used to iteratively minimise errors by adjusting the parameters of the intrinsic orientation, extrinsic orientation and the distortion to minimise

8.1.3 **8.1.3.**

$$\sum_{i=1} (x_{Ii} - x_{Pi})^2 + \sum_{i=1} (y_{Ii} - y_{Pi})^2$$

8.2 Calibration Data

Coplanar calibration (full optimization)

camera type: Photometrics Star I

data file: my_cam_cd15.dat (15 points)

f = 27.863610 [mm]

kappa1 = 4.079597e-03 [1/mm^2]

Tx = 1277.248739, Ty = -10249.790574, Tz = 206.691749 [mm]

Rx = 74.397725, Ry = -5.053187, Rz = 88.855410 [deg]

R
0.019898 -0.270599 0.962486
0.995915 -0.079445 -0.042925
0.088080 0.959408 0.267913

sx = 1.000000
Cx = 190.048502, Cy = 346.769766 [pixels]

Tz / f = 7.417982

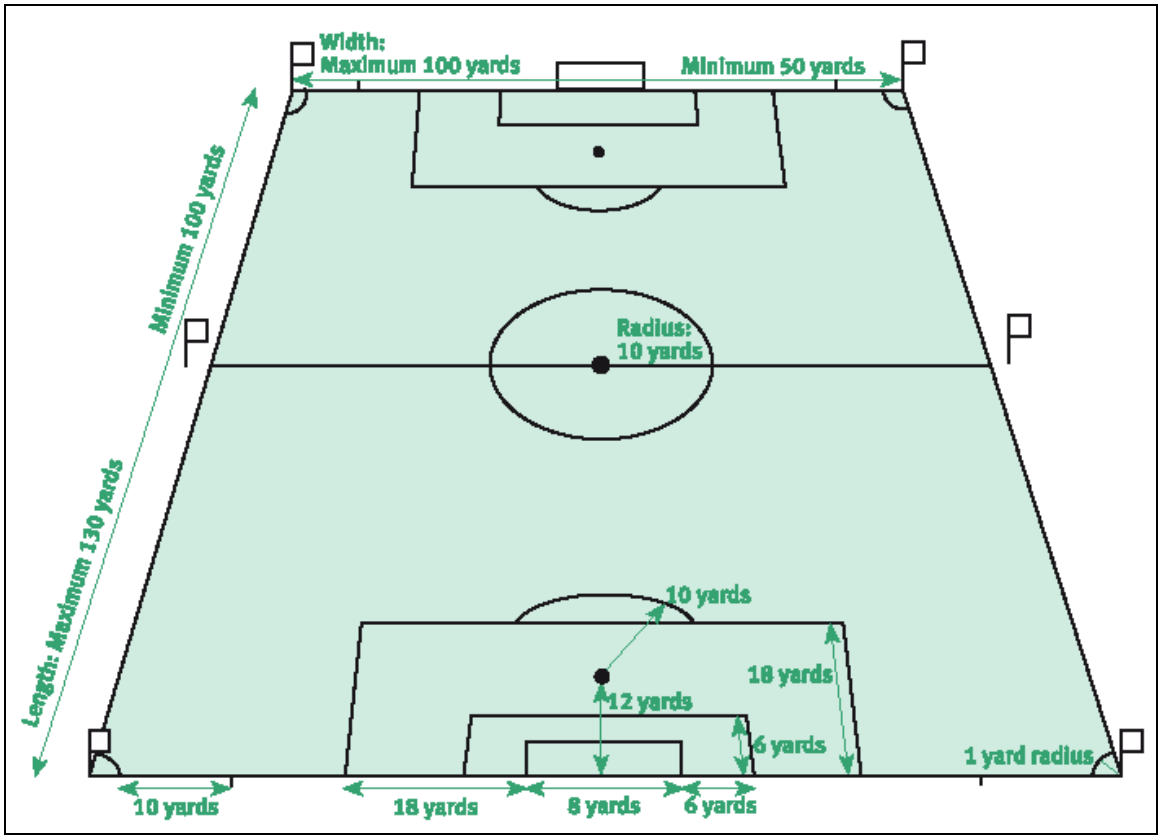
distorted image plane error:
mean = 4.233089, stddev = 2.796278, max = 11.886343 [pix], sse = 378.254005 [pix^2]

undistorted image plane error:
mean = 4.812336, stddev = 2.854931, max = 12.272334 [pix], sse = 461.487522 [pix^2]

object space error:
mean = 32.201556, stddev = 17.749751, max = 71.349420 [mm], sse = 19964.854022 [mm^2]

normalized calibration error: 11.787769

8.3 Pitch Measurements


Width:
Maximum 100 yards
Minimum 50 yards
Minimum 100 yards
Length: Maximum 130 yards
Radius:
10 yards
10 yards
18 yards
12 yards
6 yards
1 yard radius
10 yards
18 yards
8 yards
6 yards