Automated tracking of swimmers in the clean swimming phase of a race

HUDSON, Christopher <http://orcid.org/0000-0003-4698-8336>

Automated tracking of swimmers in the clean swimming phase of a race

Christopher Robert Hudson

A thesis submitted in partial fulfilment of the requirements of
Sheffield Hallam University
for the degree of Doctor of Philosophy

February 2015

Collaborating Organisation: British Swimming

Abstract

The current advice for a sports analyst when filming a large performance area is to use multiple fixed cameras or a single panning one. Neither of these options is ideal: multiple cameras must be positioned, have their shutters synchronised and their footage combined for analysis; a panning camera makes it difficult to determine an athlete's movement relative to an external frame of reference. The aim of this study was to establish a process that enabled the confident, accurate and precise use of a wide field of view for measuring distance and speed in large performance areas. Swimming was used as an example sport as it had a large performance area, which measured 50 m by 25 m.

A process for determining the accuracy and precision with which distance and speed could be reconstructed from a wide field of view was developed. A nonlinear calibration procedure was used to account for radial distortion. The Root Mean Square Error (RMSE) of reconstructed distances for a wide field of view was 16 x 10^{-3} m. This compared favourably with a three camera system reported in the literature, which had an RMSE of 46 x 10^{-3} m. In addition, it was shown that a wide field of view could be used to identify a 1% enhancement in speed when it was measured over 10 m or more.

A wide field of view was used to capture video footage of a swimming competition. This was used to track swimmers using two methods: manual and automated. The two methods showed good agreement for mean speed, but the automated one had higher variability in instantaneous speed than did the manual.

Keywords: sport analysis, calibration, wide field of view

Acknowledgements

I would like to thank Dr Simon Goodwill, Dr John Kelley and Professor Steve Haake for supervising this study.

I am also grateful to British Swimming for their sponsorship of this study and for providing me with experience of competition performance analysis.

Contents

List of figures

List of tables

1 Introduction

1.1 Competitive swimming

Competitive swimming has a history that goes back at least two hundred years (FINA 2015a). Today, the sport is regulated by FINA, the Fédération Internationale de Natation, and has been a part of every summer Olympic Games since 1896 (IOC 2015). In Great Britain, the national governing body for competitive swimming is British Swimming.

Over the last forty-five years the performance of swimmers in competitions, such as the Olympic Games and the FINA World Championships, has been analysed. As shown in Figure 1.1, it is now common for national swimming teams to film and analyse their swimmers at major swimming competitions.

Figure 1.1. Filming area for the national swimming teams at the 2011 FINA World Championships and the 2012 Olympic Games.

Such analysis aims to provide objective measures of a swimmer’s performance in four phases of a race: a start, turns, a finish, and clean swimming in between these. This data is calculated by hand: video footage of the race is replayed post-race and the swimmer is manually digitised at key distances, such as where they surfaced after the start. Coaches use the analysis data to, for example, guide training or establish race tactics.

The current analysis method relies on manual digitisation. Typically, two or three key distances are digitised per swimmer per lap (Smith, Norris and Hogg 2002). Each race is one to thirty laps and has eight swimmers in it. In addition, there are likely to be over a hundred races in each competition. Such digitisation requires a large amount of manpower. For example, Mason and Cossor (2000) reported that forty-one people worked six hour shifts around the clock to analyse the swimmers at one competition.

1.2 Automated digitisation

An obvious way of reducing the amount of manual digitisation is to automate it in some way. Digital image processing techniques are one way of doing this. These are used in video analysis tools, such as Dartfish, to track distinctive coloured objects in video footage. An example is shown in Figure 1.2. A swimming pool is a highly dynamic environment that contains movement such as ripples, splashes and reflections as well as the swimmers. Dartfish is unlikely to perform as well as a bespoke algorithm which has been optimised to account for such movements.

How digital image processing techniques could be used to automate swimming competition performance analysis is a central theme of the work reported in this document. The application of these techniques typically requires a standardised filming technique.

Figure 1.2. Digital image processing techniques are used to reduce the amount of manual digitisation that is needed. Here, a golf club head is semi-automatically tracked in Dartfish (2015).

1.3 Filming techniques

Two main filming techniques have been used in competition performance analysis. One uses multiple fixed cameras and the other a single panning camera. Neither is ideal for automated digitisation.

A single camera that has a fixed view of the whole swimming pool could be an elegant solution. It would require less equipment than a multi-camera system and as a result should be easier and quicker to setup and calibrate. In addition, only a single source of video footage would need to be calibrated and processed. The size of a swimming pool means that such a camera would have to be in a non-ideal position (i.e. with the optical axis not perpendicular to the swimming direction) or use a wide-angled lens, which is likely to result in highly distorted footage. These two options are shown in Figure 1.3.

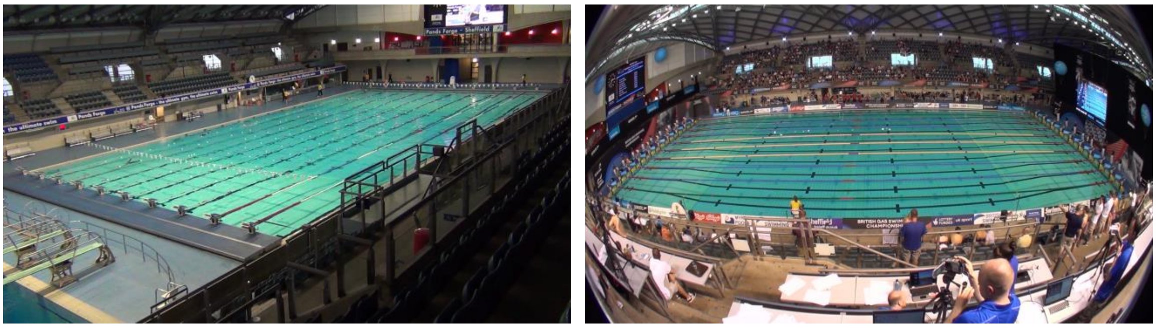


Figure 1.3. Two views of a swimming pool captured by a single fixed camera.

How a single fixed camera could be used in swimming competition performance analysis is another central theme of the work reported in this document.

1.4 Error

The use of automated digitisation and a single fixed camera are departures from the norm. They should make competition performance analysis easier and quicker, but it is reasonable to ask whether this convenience has a cost. A key question is what impact they will have on the accuracy and precision of the analysis data. For example, can an automated system digitise a swimmer with the same accuracy and precision as an experienced analyst?

The accuracy and precision of the analysis data calculated by automated digitisation and the view from a single fixed camera is the final theme of the work reported in this document.

2 Literature review

2.1 Competitive swimming

FINA regulates the size and configuration of competition pools. This is laid out in FINA (2013a). The key measurement is the swimming distance, which must be 50 m plus 0.03 m and minus 0.00 m. In addition, FINA mandate that the pool should consist of ten 2.5 m wide swimming lanes and be up to 3 metres deep. Figure 2.1 illustrates these dimensions. The lanes must be separated by lane ropes, which consist of hundreds of coloured floats. FINA stipulates that the floats in the first and last 5 m must be red and that there must be distinctive coloured floats at 15 m and 25 m in the swimming direction.

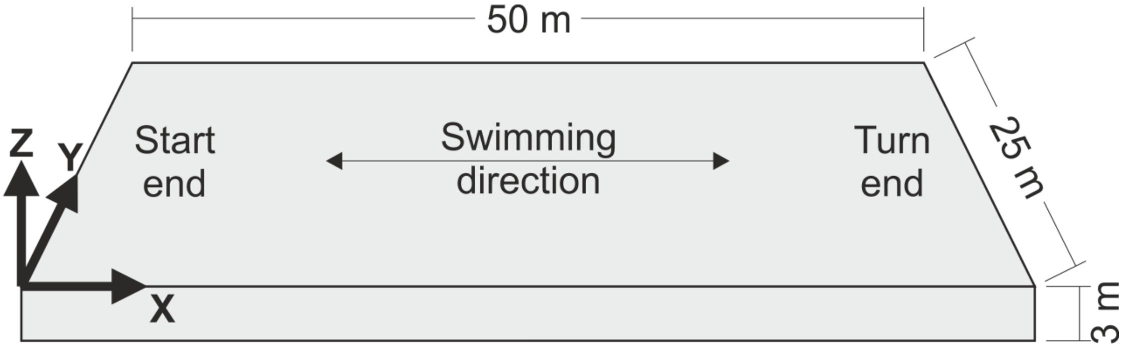


Figure 2.1. A typical 50 m competition pool. Swimming direction is along the X axis. The start end is at X = 0 m and the turn end is at X = 50 m.

There are four competitive swimming strokes: backstroke, breaststroke, butterfly and freestyle. FINA (2013b) describes the swimmer actions that are permissible in each stroke. In most races the swimmer uses one of these strokes, but in the individual medley, all four are used in the order: butterfly, backstroke, breaststroke and freestyle. Race distances are from 50 m to 1500 m, i.e. 1 to 30 laps. These are summarised in Table 2.1. At the Olympics, the only 50 m race is a freestyle one, women do not swim the 1500 m freestyle and men do not swim the 800 m freestyle.

Table 2.1. Race distances by stroke.

Stroke	Race distance (m)					
	50	100	200	400	800	1500
Backstroke	✓	✓	✓			
Breaststroke	✓	✓	✓			
Butterfly	✓	✓	✓			
Freestyle	✓	✓	✓	✓	✓	✓
Individual medley			✓	✓		

The time taken to complete the race distance is used to place the swimmers in a race. This time is referred to as the race time. An automated timing system is used: the race is started with a visual and audible start signal and time is measured at the end of each lap by pads, which the swimmer must touch to register a time.

At a competition, each race usually has a heat, a semi-final and a final. In races of 400 m and over, there is only a heat and final. In each round, the swimmers with the smallest race times qualify for the next round. There are typically sixteen swimmers in the semi-finals and eight swimmers in a final. The finalists with the three smallest race times in the final win the medals.

2.2 Competition performance analysis

2.2.1 Introduction

Over the last forty-five years the performance of swimmers in competitions has been analysed. Smith, Norris and Hogg (2002), in their review of swimming competition performance analysis, identified that the aims of such analysis are:

- To identify strengths and weaknesses in a swimmer’s performance.

- To check if a swimmer followed their race plan.
- To provide data on a swimmer’s competitor.
- To guide a swimmer’s training.

As a result, they highlight that the analysis is of use in competition and in training.

Competition performance analysis usually involves capturing video footage of a race and post-race analysis of this footage. Filming positions are usually in the spectator stands; an example is shown in Figure 2.2. An above-water view of the racing is captured. This is replayed in coordinate digitiser software post-race and the swimmers are manually digitised at key distances, e.g. on completion of the start or just prior to a turn. Time at these distances is taken from the camera’s frame rate or an external timing system. On completion of the analysis, the data is provided to coaches in a report or a spreadsheet.

Figure 2.2. Competition performance analysis filming position at the 2012 Olympic Games.

2.2.2 Race phases

An analysis of a race breaks it down into phases. These are: a start, turns, a finish, and clean swimming in between the start, turns and finish. Two ways of identifying the boundaries between phases have been used: fixed distance and individual distance.

Fixed distance is the predominant approach. The boundaries between phases are at set distances in the swimming direction. Many different set distances have been used. Currently, British Swimming use the distances described by Smith, Norris and Hogg (2002) and illustrated in Figure 2.3. One advantage of the fixed distance approach is the ease with which races can be compared. Furthermore, as will be shown in Section 2.3, simple methods can be used to identify when a swimmer is at one of the distances.

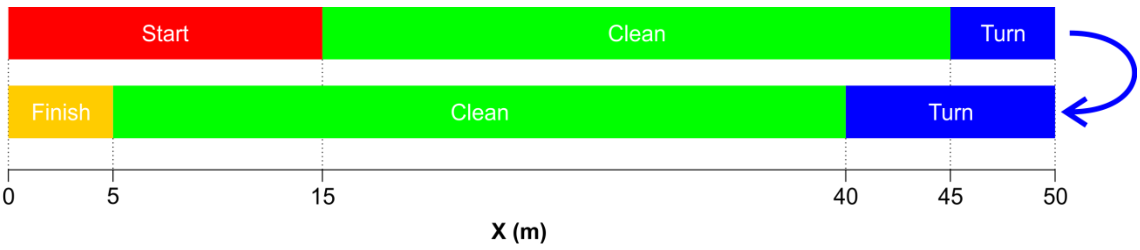


Figure 2.3. Fixed distance approach to splitting a 100 m race into phases. Boundaries between contiguous race phases are at fixed distances in the swimming direction, i.e. in the X axis.

The individual distance approach used swimmer actions to identify phase boundaries. Commonly used actions were the first hand entry (Pai, Hay and Wilson 1984) or the first head emersion (Veiga et al. 2012) after a period of underwater swimming. The analysis calculated the distance on the X axis at which these actions occurred. This gave a more accurate delineation of race phases than the fixed distance approach, but required a more complicated method of measuring distance.

2.2.3 Clean swimming phase

There is evidence that clean swimming is the most important race phase. Table 2.2 shows that the clean swimming phase takes up most of the race distance. In addition, it accounts for most of the race time. This is because it is the longest and it is swum at the slowest speed. For example, Veiga et al. (2012) found that the mean speed in the start, turn and clean swimming phases were 2.010 m s^{-1}, 1.685 m s^{-1} and 1.560 m s^{-1} (respectively).

Table 2.2. Length of race phases. The fixed distance boundaries illustrated in Figure 2.3 were used to do these calculations.

Phase	50	100	200	400	800	1500
Start (m)	15	15	15	15	15	15
Turns (m)	0	15	45	105	225	435
Finish (m)	5	5	5	5	5	5
Free swim (m)	30	65	135	275	555	1045
Free swim (%)	60.0	65.0	67.5	68.8	69.4	69.7

The time taken to swim the clean swimming phase, which is typically expressed as the speed in this phase, has been shown to have the highest correlation with race time. Three studies have analysed all race phases and correlated the time or speed in the phase with race time: Arellano et al. (1994); Mason and Cossor (2000); Thompson Haljand and MacLaren (2000). In all races except one, clean swimming speed had either the highest or second highest correlation with race time. Furthermore, it had the highest in over 80% of the races. So, performance in the clean swimming phase accounted for most of the variability in race time.

2.2.4 Conclusion

This section reviewed competition performance analysis and found:

- It aims to give coaches objective data for use in competition and training.

- The data is calculated by manually digitising the swimmer in above-water video footage.
- A race is split into four phases: a start, turns, a finish, and clean swimming in-between these. The boundary between phases is at a fixed distance or a predetermined swimmer action.
- Of the four phases, clean swimming is the longest. In addition, the time or speed in this phase has been found to contribute the most to the overall performance.

In the next section, the focus switches to the methods that have been used to measure distances in competition performance analysis. Due to its importance, a particular emphasis is placed on the clean swimming phase. Then, in section 2.4, approaches to automating the digitisation of swimmers are reviewed. This is done with a view to how the manual digitisation methods currently used in competition performance analysis could be automated.

2.3 Measuring distance

2.3.1 Introduction

A measurement of distance from video footage requires a way of transforming between pixel and world coordinate systems. The pixel coordinate system is used to define the index of a picture element (or pixel) in an image, which is a 2D matrix of pixels. The pixel coordinate system has U and V axes. The world coordinate system represents 3D points and distances in the real world and has X, Y and Z axes. The transformation from world to pixel coordinate systems is called projection and the reverse is called reconstruction.

Projection and reconstruction are typically performed by a calibration model. A calibration model is calculated by a calibration procedure. These procedures define how to calculate the relationship between pixel and world coordinate systems and the equations that are used to do projection and reconstruction. In some calibration procedures the relationship between pixel and world coordinate systems is only valid for a given plane in the world coordinate system. This is called the calibrated plane.

Table 2.3 lists the methods that have been used to measure distance in competition performance analysis. These methods and a calibration procedure called nonlinear, which has been shown to be applicable to the wide-angle viewing conditions typical of sporting competitions, are discussed in the following subsections.

Table 2.3. Methods used to measure distance in competition performance analysis.

Method	Study
Lap time	Craig and Pendergast (1979) Craig et al. (1985) Chollet et al. (1996) Pelayo et al. (1996) Chollet et al. (1997) Chollet and Pelayo (1999)
Panning	British Swimming Kennedy et al. (1990)
Optical axis	Wakayoshi et al. (1992) Chollet and Pelayo (1999) Thompson et al. (2000) Thompson, Haljand and MacLaren (2000) Chollet, Tourny-Chollet and Hogie (2001) Tourny-Chollet et al. (2002) Takagi et al. (2004) Hellard et al. (2008)
Linear scaling	East (1970) Chow et al. (1984) Miller, Hay and Wilson (1984) Pai, Hay and Wilson (1984) Chengalur and Brown (1992) Arellano et al. (1994) Pelayo et al. (1996) Arellano (2001) Chatard et al. (2001a) Chatard et al. (2001b) Chatard et al. (2001c) Chatard et al. (2001d) Chatard et al. (2001e) Cossor and Mason (2001) Girold et al. (2001a) Girold et al. (2001b) Ikuta, Mason and Cossor (2001) Mason and Cossor (2001) Riewald (2001) Wilson (2001) Veiga et al. (2012)
2D-DLT	Veiga et al. (2012)
Unknown	Mason, Loschner and Fowlie (1995) Mason and Cossor (2000) Huot-Marchand et al. (2005)

2.3.2 Lap time

This is not a calibration procedure. It does not even require video footage. Instead, the lap times from the official timing system are used to calculate an average speed for the entire lap as 50 / *t*, where *t* is the lap time in seconds.

Examples

Craig and Pendergast (1979) pioneered this method at the 1976 US Olympic Trials. Time to swim each 100 m of a race was obtained from the automated timing system. Clean swimming speed was calculated from this. Stroke rate was calculated in real-time by three observers: each used a stopwatch to time five consecutive stroke cycles in each lap. Speed and stroke rate data was used to construct "stroke rate-velocity curves" (Craig and Pendergast 1979, p279), which described the relationship between speed, stroke rate and stroke length. An example is shown in Figure 2.4.

Figure 2.4. Stroke rate-velocity curves for the freestyle swimmers at the 1976 US Olympic Trials. Stroke length is the gradient of the line drawn from the origin to a point on a curve. The "2 fastest males" were national record holders. Adapted from Craig and Pendergast (1979, p280).

Chollet et al. (1996 and 1997) built on the work done at the US Olympic Trials in 1976. A key development was the calculation of stroke rate from video footage instead of real-time observations. Four panning cameras were used: each camera followed two of the eight swimmers in each race.

Advantages

The lap time method has the following advantages:

- Automated. The timing data is collected by the automated timing system and is often available in a machine-readable file format, e.g. LENEX (OMEGA Timing 2015). As a result, calculation of average speed for a lap can be largely automated.
- High accuracy and precision data source. The timing data is the official result of the race: it is this data that determines, for example, a world record or the winner on an Olympic medal. In addition, the calculation of speed from time is straightforward and can be automated. Therefore, there is little chance for errors to occur.

Disadvantages

The lap time method has the following disadvantages:

- Average swimming speed for a lap is only an estimate of clean swimming speed. All studies that have used this method acknowledge that it leads to an overestimation of clean swimming speed (Craig and Pendergast 1979, Craig et al. 1985, and Chollet et al. 1997).
- May be unsuitable for elite swimmers. Enhancements as small as 1% can result in a medal-winning performance at a major competition (Pyne,

Trewin and Hopkins 2004). An estimate of clean swimming speed may be unable to reliably identify such enhancements.

2.3.3 Panning camera

This calibration procedure used panning, instead of fixed cameras. Each camera followed one or more swimmers in each race and distances were measured with reference to the lane ropes.

Examples

Kennedy et al. (1990) used panning cameras at the 1988 Olympic Games. A single camera was used to capture video footage of the eight swimmers in each race. The phase boundaries were at distances that coincided with a change in colour of the floats on the lane ropes. These were at 15 m, 40 m and 45 m in the X axis. During post-race analysis, transverse lines were visually aligned with the appropriate float on each lane rope. This had to be done for each frame in the footage in which the analyst thought the swimmer was at a phase boundary.

British Swimming also uses panning cameras. However, unlike Kennedy et al. (1990), they can measure distances that do not coincide with a change in float colour. The measurements are done by way of a map, which is constructed using the following steps:

1. Adjust the lane ropes. Make sure there are no gaps between floats and that the corresponding change in float colour on each lane rope is at the same distance in the X axis.
2. Identify the floats at 15 m, 25 m, 35 m and 45 m in the X axis. Also, count the number of floats in each coloured section of floats.

3. Construct the map. Calculate the average metres per float between 0 m and 15 m, 15 m and 25 m, 25 m and 35 m, 35 m and 45 m, and 45 m and 50 m in the X axis. From this, construct a diagram of the lane rope, which shows the colour and distance in the X axis of the floats near the phase boundaries.

During analysis the float that the swimmer is aligned with is identified. This is done by drawing a transverse line between corresponding floats on two or more lane ropes. The number of floats between the identified one and the nearest change in float colour is then counted. The distance of the identified float can then be found in the map.

Advantages

The panning calibration procedure has the following advantages:

- Freely positioned camera. The camera can be located wherever it can capture the required video footage.
- Multiple measurements per camera. A single camera can be used to measure more than one fixed distance or the distance between swimmer actions.

Disadvantages

The panning calibration procedure has the following disadvantages:

- Non-fixed view. Each frame in the video footage had a different view of the calibrated plane, i.e. the lane ropes. This led to a reconstruction process that was more time-consuming than it would have been for a fixed view camera.

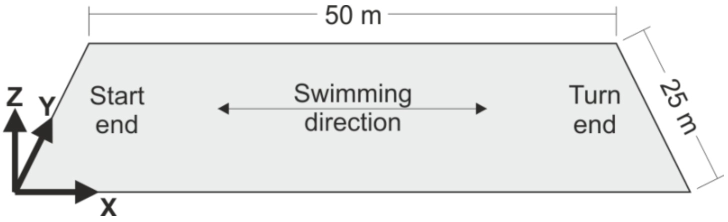
- Potential for out-of-plane measurements. Consider the case of digitising a swimmer's head at a fixed distance in a breaststroke race. In this stroke the swimmer's head is on the water's surface at two instants in each stroke cycle. Each cycle is likely to have a duration of one second or more (according to, e.g., East 1970; Pai, Hay and Wilson 1984; and Kennedy et al. 1990). As a result, the head is unlikely to be on the water's surface at each fixed distance and so measurements off the calibrated plane would probably occur. However, Kennedy et al. (1990, p190) stated that the error in such measurements were likely to be "small and primarily random in nature".

2.3.4 Optical axis

The optical axis calibration procedure is used to measure the time that a swimmer is at a fixed distance boundary between two race phases. Table 2.4 summarises the distances these have been at. Two studies did not describe this: Chollet and Pelayo (1999) had two cameras near 25 m and Takagi et al. (2004) used five cameras.

Table 2.4. Phase boundaries for the studies that used the optical axis calibration procedure.

Study	X (m)
Wakayoshi et al. (1992)	5, 10, 45
Thompson et al. (2000)	5, 7.5, 15, 25, 42.5
Thompson, Haljand and MacLaren (2000)	5, 7.5, 15, 25, 42.5
Chollet, Tourny-Chollet and Hogie (2001)	7.5, 42.5
Tourny-Chollet et al. (2002)	7.5, 42.5
Hellard et al. (2008)	7.5, 15, 25, 42.5



For this calibration procedure, each camera is positioned at the required distance and its optical axis is set perpendicular to the swimming direction. The time at which the swimmer intersects the optical axis gives their time at that distance. In this way, each camera acts like a video-based timing gate.

Examples

Wakayoshi et al. (1992) analysed the performance of swimmers at the 1988 Japanese Olympic Trials and the 1989 Pan Pacific Championships. As shown in Figure 2.5, three cameras used the optical axis calibration procedure and a fourth camera (i.e. Camera 3) was used to calculate stroke rate. An external time synchronisation system was used; as the start signal light could be seen by one camera (i.e. Camera 1), race and video time could be harmonised in the footage from all cameras.

Figure 2.5. Camera setup at 1988 Japanese Olympic Trials and 1989 Pan Pacific Championships (Wakayoshi et al. 1992, p136).

Thompson, Haljand and MacLaren (2000) describe the way this calibration procedure is used:

1. Mark control points on the pool deck at the fixed distances (i.e. 5 m, 10 m and 45 m in the X axis for Wakayoshi et al. 1992).
2. Position and orientate each camera so its optical axis intersects the control points.
3. Capture footage of the control points from each camera.
4. Without changing the cameras' position or orientation, capture footage of the races.
5. In the coordinate digitiser, superimpose transverse lines between the control points at each fixed distance.
6. During analysis, count the number of times the synchronisation signal occurs between the start pistol firing and the instant the swimmer intersects a transverse line. Convert this count to seconds using the known frequency of the synchronisation signal.
7. Calculate clean swimming speed as d / t, where d is the distance between two cameras in the X axis and t the time taken to swim it.

Wakayoshi et al. (1992) calculated speed between Camera 2 and Camera 4 on the first lap and Cameras 1 and 4 on subsequent ones. So, d was either 35 m or 40 m in the X axis. Other studies have used shorter distances, e.g. Chollet and Pelayo (1999) used 15 m in the X axis.

Advantages

The optical axis calibration procedure has the following advantages:

- Simplicity. It has a relatively simple set-up procedure and projection and reconstruction are straightforward. So, it is easy to implement and understand.

- High accuracy and precision. This simplicity suggests that there are fewer chances for errors to occur. Furthermore, there is no parallax error as the swimmers are always measured on the optical axis.

Disadvantages

The optical axis calibration procedure has the following disadvantages:

- Each camera can only measure time at one distance. As a result, at least two cameras are needed to measure clean swimming speed but more are typically used. This can cause logistical problems, e.g. synchronisation cabling has to be run between cameras that are up to 40 m apart.
- Cameras cannot be freely positioned. If a required camera location is not available, then time at that distance cannot be measured. Anecdotally, this is a common occurrence: typically, at least one camera location is not available due to e.g. a spectator exit route, spectator seating or occupation by competition staff such as commentators or the media.

2.3.5 Linear scaling

The linear scaling calibration procedure extends the optical axis one. The key addition is a linear scaling coefficient, which describes the relationship between the pixel and world coordinate systems. This can be used to construct a calibrated plane or do reconstructions or projections off the optical axis.

The scaling coefficient is calculated from the distance, in metres and pixels, between control points on the pool deck or the lane ropes. It is used to do reconstructions and projections, for which the equations are:

$$X = sU \tag{2.1}$$

$$U = \frac{1}{s}X \tag{2.2}$$

where s is the scaling coefficient in metres per pixel, X is the distance in the world coordinate system and U the distance in the pixel coordinate system.

Examples

Pai et al. (1984) measured the clean swimming speed of the swimmers at the 1982 Commonwealth Games. Four control points on the near and far edges of the pool deck were used to calculate a calibration model. This gave a calibrated plane on the water surface that was 6.48 m in the X axis. This is shown in Figure 2.6. The calibrated plane was mid-pool, i.e. close to 25 m in the X axis.

Figure 2.6. Calibrated plane used at the 1982 Commonwealth Games (Pai et al. 1984, p228). L and l_i (i = 1 to 9) were 6.48 m. The length of each l_i in pixels was measured and used to calculate scale factor s for each lane.

The calibrated plane allowed Pai et al. (1984) to measure the distance at which the swimmer performed a certain action. The chosen action was when the tip of the swimmer’s index finger intersected the water surface at the start and end of a stroke cycle. This ensured that the measured distance was on the calibrated plane.

Arellano et al. (1994) positioned control points on the lane ropes instead of the pool deck. Cameras were located to capture footage of 10 m from the start end, the central 30 m, and 7.5 m at the turn end. The central camera was used to calculate stroke rate and the ones at the start and turn ends measured phase boundaries, which were at 7.5 m, 10 m, 40 m and 42.5 m in the X axis. Implicit in this is that the cameras at the start and turn end had to each measure two fixed distances, i.e. 7.5 m and 10 m from the end of the pool. The following steps were used to do this:

1. In the video footage, identify the float on the lane rope at 5 m and 10 m from both ends of the pool. It was known that the floats changed in colour every 5 m.
2. Determine the length of these 5 m sections of lane rope in pixels at the near and far side of the pool.
3. Calculate scaling coefficients for the near and far sides of the pool.
4. Use Equation 2.2 to project transverse lines at 7.5 m.

Steps 1 to 4 are illustrated in Figure 2.7.

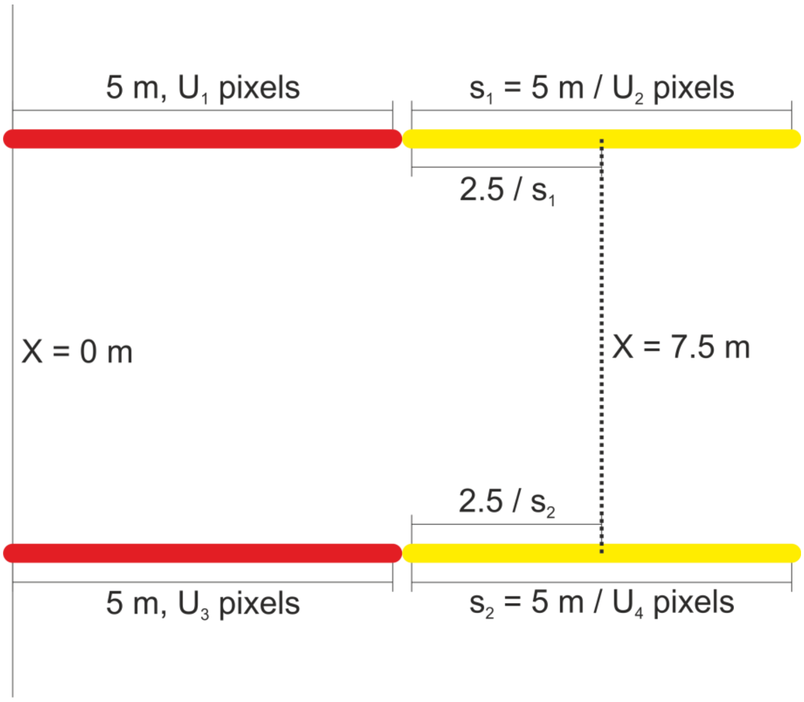


Figure 2.7. Projection of a transverse line using the linear scaling calibration procedure and control points on the lane ropes. Scale coefficients (s_1 and s_2) are calculated for the nearest and furthest lane ropes. The transverse line at X = 7.5 m is drawn between these lane ropes from: $U_1 + (2.5 / s_1)$ pixels to $U_3 + (2.5 / s_2)$ pixels.

Advantages

The linear scaling calibration procedure has the following advantages:

- Simplicity. The calculation of the scaling coefficient and equations for reconstruction and projection are straightforward to understand and implement.
- Multiple measurements per camera. Each camera can measure more than one distance.

Disadvantages

The linear scaling calibration procedure has the following disadvantages:

- Small calibrated plane. The calibrated planes were less than 10 m in the X axis. So, two cameras would be needed to measure time or speed throughout the clean swimming phase.
- Potential for out-of-plane measurements. This was described in the **Panning camera** subsection.

2.3.6 2D-DLT

The 2D-DLT calibration procedure (Walton 1981) is the 2D equivalent of 3D-DLT (Abdel-Aziz and Karara 1971). A 2D-DLT calibration model requires the same inputs as linear scaling, i.e. the world and pixel coordinates of control points. A system of linear equations is formed from these and solved to calculate eight DLT parameters, A_i (i = 1 to 8). If more than four control points are provided, then the system of equations is overdetermined and a least squares solution is calculated. Once calculated, the DLT parameters are used to do projection and reconstruction.

Projection

Kwon (1999, p397) states that Walton's projection equations are:

$$U = \frac{A_1 X + A_2 Y + A_3}{A_7 X + A_8 Y + 1} \qquad 2.3$$

$$V = \frac{A_4 X + A_5 Y + A_6}{A_7 X + A_8 Y + 1} \qquad 2.4$$

where A_i (i = 1 to 8) are the DLT parameters, U and V are the pixels coordinates, and X and Y the world coordinates.

Reconstruction

Reconstruction is achieved by rearranging Equations 2.3 and 2.4 into a pair of linear equations, which are then solved for X and Y:

$$X(A_1 - A_7U) + Y(A_2 - A_8U) = U - A_3 \tag{2.5}$$

$$X(A_4 - A_7V) + Y(A_5 - A_8V) = V - A_6 \tag{2.6}$$

where A_i (i = 1 to 8) are the DLT parameters, U and V are the pixels coordinates, and X and Y the world coordinates.

Examples

Veiga et al. (2012) measured the clean swimming speed of the swimmers at an international competition. Three video cameras were used to film the races and a 2D-DLT calibration model was calculated for each camera's footage. Eight control points on the pool deck were used to calculate A_i (i = 1 to 8). Calibrated planes were on the water surface and about 15 m in the X axis. The calibration model error (as defined in Equation 3.1) for the distances between 32 points on the lane ropes was 46 x 10^{-3} m.

Advantages

The 2D-DLT calibration procedure has the following advantages:

- Freely positioned camera. The camera can be located wherever it can capture the required video footage.
- Multiple measurements per camera. Each camera can measure more than one fixed distance or the distance between swimmer actions.
- Error minimisation. If more than four control points are used, then a least square solution for A_i (i = 1 to 8) is calculated. This should go some way

to reducing the projection and reconstruction errors that are caused by errors in measuring a control point's world or pixel coordinates. Contrast this with the linear scaling procedure: any error in a control point's coordinates would propagate to the scaling coefficient and hence projection and reconstruction.

- Closed-form solution. The projection and reconstruction equations (2.3 to 2.6) are closed-form expressions. Such expressions have a low computational burden (Heikkilä and Silvén 1997).
- Control points are prescribed by the swimming pool's geometry.

Disadvantages

The 2D-DLT calibration procedure has the following disadvantages:

- Complexity. Compared to optical axis and linear scaling, 2D-DLT is a relatively complex calibration procedure. The DLT parameters, A_i (i = 1 to 8), are not easily understood as they are not based on physical parameters such as focal length (Heikkilä and Silvén 1997). In addition, although they are closed-form expressions, the projection and reconstruction equations (i.e. Equations 2.3 to 2.6) are harder to understand and solve than, for example, the linear scaling ones.
- Assumption of a linear relationship between pixels and metres. Tsai (1987) states that the 2D-DLT calibration procedure (as described in this subsection) needs extending to account for nonlinear lens distortions. This may not be needed if a narrow field of view is used. Overall, it probably means that more than one camera would be required, which has the logistical problems identified in the **Optical axis** subsection.

- Potential for out-of-plane measurements. This was described in the **Panning camera** subsection.

2.3.7 Nonlinear

The nonlinear calibration procedure uses two steps to calculate the relationship between pixel and world coordinate systems. The first accounts for the effects that the camera’s lens and the imperfections in the camera’s construction has on the video footage that it captured. This stage uses video footage of a planar calibration object; images from this are shown in Figure 2.8. The second step estimates the camera’s pose, i.e. its position and orientation in the world coordinate system. The world and pixel coordinates of four or more control points are required for this.

Figure 2.8. Images from the video footage of a planar calibration object (Bouguet 2013).

A popular and convenient implementation of the nonlinear calibration procedure is provided in the form of a MATLAB toolbox (Bouguet 2013). This toolbox implements the methods described in two studies: Heikkilä and Silvén (1997) and Zhang (2000). Another implementation is OpenCV (OpenCV 2015). Knowledge of C++ programming is required to use this so it is less convenient than the MATLAB toolbox.

The following steps are used to calculate a nonlinear calibration model using the MATLAB toolbox:

1. Set and lock the video camera's settings. Any external lens (if used) must be fitted and zoom and focus fixed. However, exposure time does not have to be locked.
2. Capture video footage of a calibration object. A convenient choice is a checkerboard pattern attached to a flat surface (Zhang 2000). The calibration object should be held in a range of positions and orientations such that the object appears in each part of the field of view. This gives good lens coverage.
3. Select images from the footage of the calibration object. Ensure that the selected images cover the camera's field of view and show the calibration object in a range of orientations.
4. Calculate the world coordinates of the calibration object's checkerboard corners. The checkerboard geometry and size of each square is used to do this.
5. Find the pixel coordinates of the checkerboard corners in each image of the calibration object. A semi-automated and sub-pixel accurate function is used. Initial pixel coordinates of the object's four bounding corners in each image are provided by the user. Digital image processing techniques are then used to automatically determine the sub-pixel accurate pixel coordinates of all checkerboard corners in each image.
6. Calculate the calibration model's intrinsic parameters and distortion coefficients. An optimisation approach is used; this attempts to minimise reprojection error, which is the sum of the distance between found and

   projected checkerboard corners in each image of the calibration object. During this calculation, the user can choose to calculate or fix any of the intrinsic parameters or distortion coefficients. For example, if the user decides that the camera does not suffer from tangential distortion, then it can be fixed at zero.

   The intrinsic parameters are the focal length, the principal point and the skew. The distortion parameters describe two types of lens distortion: radial and tangential. These are described in more detail in the **Projection** and **Reconstruction** subsections.

7. Calculate extrinsic parameters. This requires the intrinsic parameters, distortion coefficients and the pixel and world coordinates of at least four control points. If more than four control points are provided, then the solution to the extrinsic parameters is overdetermined and an error minimisation process is undertaken.

   The extrinsic parameters consist of two matrices, which are called R and T. These are described in more detail in the **Projection** and **Reconstruction** subsections.

The nonlinear calibration model calculated by the above steps is used to do projections and reconstructions, which are a sequence of transformations between world, camera, normalised image and pixel coordinate systems.

Projection

The first transformation in the projection sequence is from world to camera coordinate systems. This is implemented as a rigid body transformation, which consists of a rotation followed by a translation. This aligns and then collocates the origins of the two coordinate systems. Figure 2.9 illustrates this.

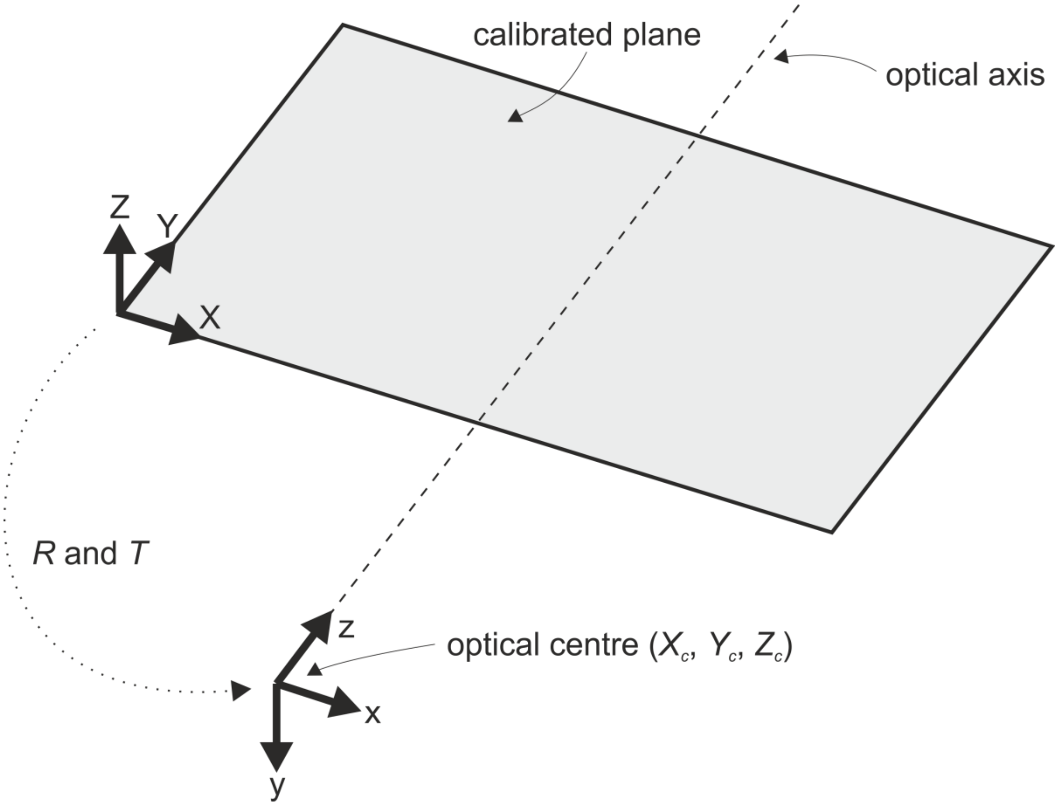


Figure 2.9. World to camera coordinate system transformation. A rotation (*R*) followed by a translation (*T*) aligns and then collocates the two coordinate systems.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + T \qquad 2.7$$

$$R = \begin{bmatrix} \cos\beta\cos\gamma + \sin\alpha\sin\beta\sin\gamma & \cos\beta\,(-\sin\gamma) + \sin\alpha\sin\beta\cos\gamma & \cos\alpha\sin\beta \\ \cos\alpha\sin\gamma & \cos\alpha\cos\gamma & -\sin\alpha \\ (-\sin\beta)\cos\gamma + \sin\alpha\cos\beta\sin\gamma & (-\sin\beta)(-\sin\gamma) + \sin\alpha\cos\beta\cos\gamma & \cos\alpha\cos\beta \end{bmatrix} \qquad 2.8$$

$$T = R \begin{bmatrix} -X_c \\ -Y_c \\ -Z_c \end{bmatrix} \qquad 2.9$$

where α, β and γ are pitch, yaw and roll rotations about the world X, Y and Z axes (respectively) and X_c, Y_c and Z_c are the coordinates of the optical centre in the world coordinate system. Rotations are applied in roll-pitch-yaw order, i.e. the world coordinates are first rotated about the Z axis, then the X axis and finally the Y axis. Positive rotations are applied clockwise when looking at the positive end of the axis.

The second transformation in the sequence is from camera to undistorted normalised image coordinate systems. This consists of a pinhole projection of points on the calibrated plane onto the image plane. During this, a focal length of unity is assumed.

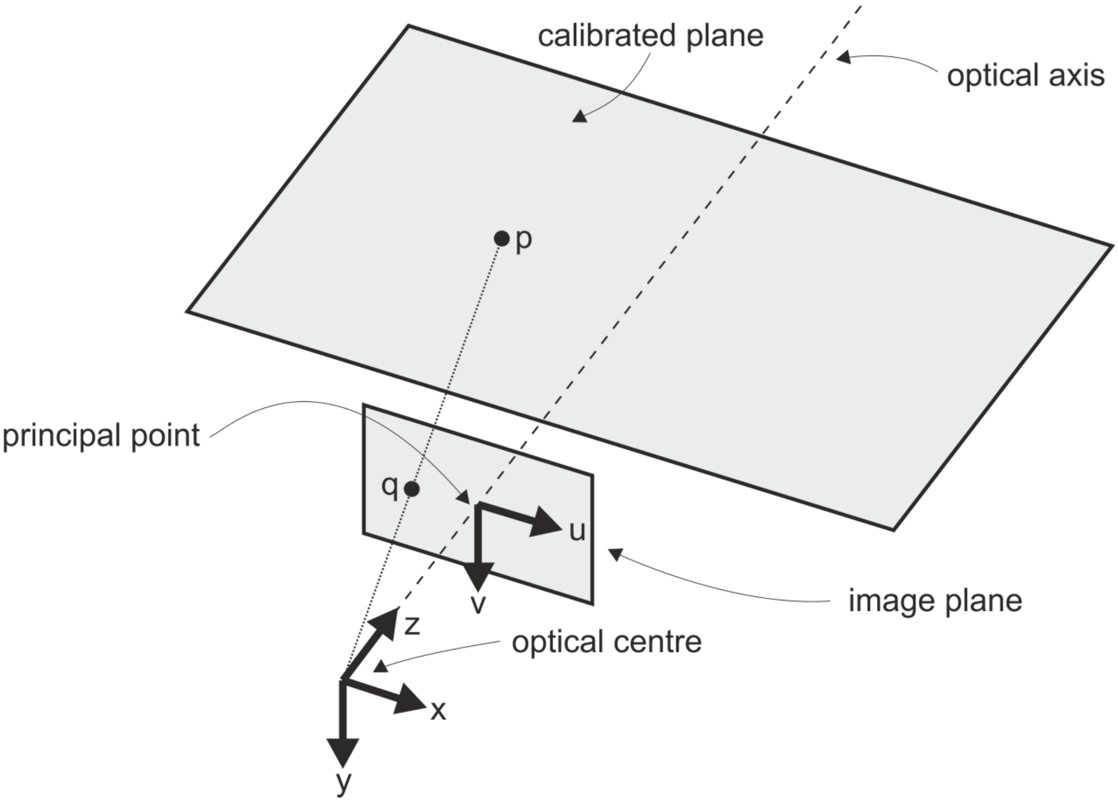


Figure 2.10. Camera to undistorted normalised image coordinate system transformation. Point p on the calibrated plane is pinhole projected to point q on to the image plane.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{z} \begin{bmatrix} x \\ y \end{bmatrix} \qquad 2.10$$

The third transformation in the sequence is from undistorted normalised image to distorted normalised image coordinate systems. This transformation applies two lens distortions: radial and tangential. Radial distortion results from differences in magnification across a lens. Barrel distortion, which is common in wide-angle and fisheye lenses, is when magnification decreases as distance from the principal point increases. Tangential distortion is caused by the misalignment of a camera's lens and image plane (Bradski and Kaehler 2008). The effect of lens distortion is points that are not the same as for a pinhole projection, i.e. Equation 2.10.

The transformation for a camera not fitted with a fisheye lens has equations that describe radial (Equation 2.12) and tangential (Equation 2.13) distortion. These are applied to the undistorted normalised image coordinates in Equation 2.11:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \left(\delta_r \begin{bmatrix} u \\ v \end{bmatrix}\right) + \begin{bmatrix} \delta_u \\ \delta_v \end{bmatrix} \tag{2.11}$$

$$\delta_r = 1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \tag{2.12}$$

$$\begin{bmatrix} \delta_u \\ \delta_v \end{bmatrix} = \begin{bmatrix} 2k_4 uv + k_5(r^2 + 2u^2) \\ 2k_5 uv + k_4(r^2 + 2v^2) \end{bmatrix} \tag{2.13}$$

where r is the magnitude of (u, v), k_i (i = 1 to 3) are radial distortion coefficients, and k_i (i = 4 to 5) are the tangential distortion coefficients.

A fisheye lens is designed to radially distort. This is achieved by including a lens with a large negative meniscus at the front of the system of lenses (Saxby 2011). This lens changes the direction of incoming light rays toward the optical axis, as shown in Figure 2.11, and causes barrel distortion.

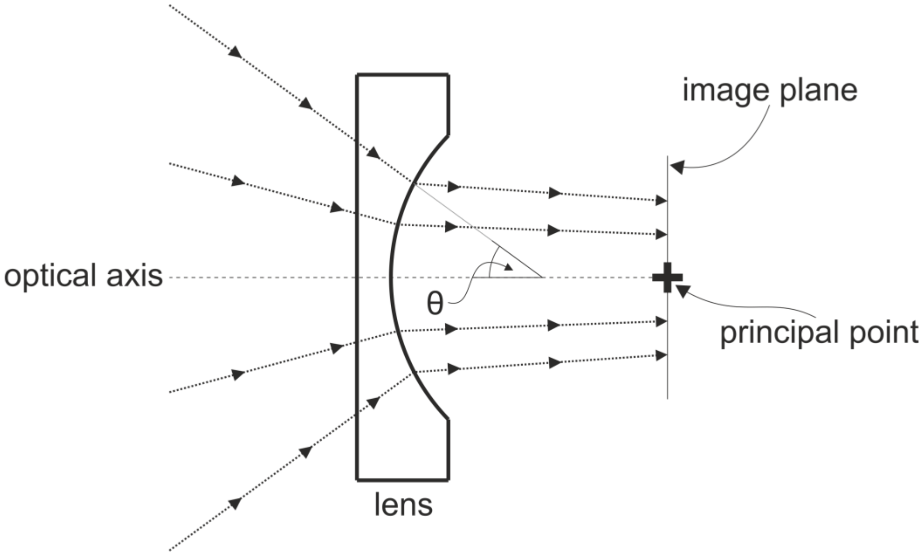


Figure 2.11. Fisheye lens. A lens with a large negative meniscus changes the direction of incoming light rays toward the optical axis. The angle between an incoming ray and the optical axis is θ.

The fisheye transformation that is implemented in Bouguet's (2013) toolkit is based on work by Kannala and Brandt (2006). Their position is that a real fisheye lens does not follow any single theoretical projection. So, they proposed that a fisheye lens is best described by a generic fisheye projection equation. This is given in Equations 2.15 and 2.16 and is applied to the undistorted normalised image coordinates in Equation 2.14:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \frac{\theta_d}{r} \begin{bmatrix} u \\ v \end{bmatrix} \qquad 2.14$$

$$\theta_d = \theta(1 + k_6\theta^2 + k_7\theta^4 + k_8\theta^6 + k_9\theta^8) \qquad 2.15$$

$$\theta = \tan^{-1} r \qquad 2.16$$

where r is the magnitude of (u, v), k_i (i = 6 to 9) are the radial distortion coefficients, and θ is the angle between an incoming ray and the optical axis.

Bouguet's (2013) toolkit does not account for tangential distortion for a fisheye lens. In contrast, Kannala and Brandt (2006) do. Therefore, Bouguet's implementation could be considered incomplete.

The fourth, and final, transformation in the sequence is from the distorted normalised image coordinate system to the pixel coordinate system. This consists of a change in origin and the effects of two physical features of the camera: focal length and skew:

$$\begin{bmatrix} U \\ V \end{bmatrix} = \begin{bmatrix} u' f_u + v' \tau + U_0 \\ v' f_v + V_0 \end{bmatrix} \tag{2.17}$$

where U_0 and V_0 are the coordinates of the principal point in pixels, f_u and f_v are the focal length on the U and V axes (in pixels), and τ is the skew between these axes.

The principal point is the centre of the radial distortion. It is assumed to be at the image centre, but imperfections in the camera may mean this is not the case. The skew is usually considered to be zero, i.e. the angle between the axes is 90° (Bouguet 2013, Bradski and Kaehler 2008 and Zhang 2000). The use of two focal lengths allows for non-square pixels. The aspect ratio, which is defined as f_v/f_u is a measure of this; its default value is one, i.e. square pixels are assumed.

Reconstruction

The first transformation is from the pixel to the distorted normalised image coordinate system. This is achieved by a re-arrangement of Equation 2.17 to give:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} (U - v'\tau - U_0)/f_u \\ (V - V_0)/f_v \end{bmatrix} \tag{2.18}$$

The second transformation in the sequence is from the distorted normalised image to the undistorted normalised image coordinate system. This transformation removes radial and tangential distortion. There isn't a closed form solution to this (Heikkilä and Silvén 1997). So, Bouguet's (2013) toolkit uses an iterative method in which an initial approximation converges toward a final value over twenty iterations.

The non-fisheye transformation is:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{\delta_r}\left(\begin{bmatrix} u' \\ v' \end{bmatrix} - \begin{bmatrix} \delta_u \\ \delta_v \end{bmatrix}\right) \tag{2.19}$$

where δ_r, δ_u and δ_v are as defined in Equations 2.12 and 2.13. Equations 2.19, 2.12 and 2.13 are iterated twenty times. On the first iteration, r is unknown, so an approximate value is used (in Equations 2.12 and 2.13): it is set to $|u', v'|$. On subsequent iterations r is set to $|u, v|$, i.e. using the u and v calculated on the previous iteration.

The fisheye transformation is:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{\tan\theta}{r'}\begin{bmatrix} u' \\ v' \end{bmatrix} \tag{2.20}$$

$$\theta = \frac{\theta_d}{1 + k_6\theta^2 + k_7\theta^4 + k_8\theta^6 + k_9\theta^8} \tag{2.21}$$

where k_i (i = 6 to 9) and θ are as defined in Equations 2.14 to 2.16 and r' is the magnitude of (u', v'). Equation 2.21 is iterated twenty times. On the first iteration, θ is unknown so an approximate value is used: it is set to $\tan^{-1} r'$. On subsequent iterations, the previously calculated value of θ (from Equation 2.21) is used.

The third transformation in the sequence is from the undistorted normalised image to the camera coordinate system. Dunn et al. (2012) established a way of doing this transformation using line-plane interception geometry.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = d \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{2.22}$$

where d is the distance, in the camera coordinate system, between the optical centre and point (x, y, z).

The fourth, and final, transformation in the sequence is from camera to world coordinate systems. This consists of rigid body transformation, which is the reverse of that performed in Equation 2.7.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = R^{-1} \left(\begin{bmatrix} x \\ y \\ z \end{bmatrix} - T \right) \tag{2.23}$$

Examples

There are no known uses of the nonlinear calibration procedure in swimming competition performance analysis. However, Dunn et al. (2012) used it to construct a calibrated plane from video footage of a tennis competition. One camera used the nonlinear calibration procedure; this camera had a field of view that included the singles court and the surrounding area (which was needed as the play was not restricted to the court). The camera was located in the spectator stands at approximately (-8 m, -12 m, 9 m) in the world coordinate system. Four control points were located at the corners of the singles court, which measured about 24 m in the X axis and 8 m in the Y axis (ITF 2015). The calibrated plane was larger; it was approximately 32 m by 18 m. Figure 2.12 shows the camera location, singles court, control points and calibrated plane.

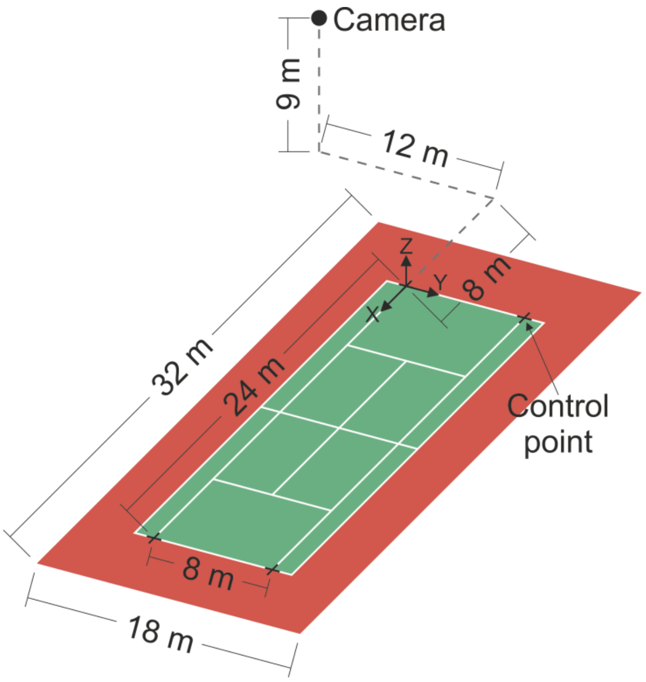


Figure 2.12. Camera location, tennis court and calibrated plane used by Dunn et al. (2012). Control points were at the corners of the singles court. The calibrated plane measured 32 m by 18 m.

Dunn et al. (2012) used a 1:30 scale model of the competition setup to assess reconstruction accuracy. The nonlinear and 2D-DLT calibration procedures were used to reconstruct 60 test points inside the singles court and 64 outside it. The results are summarised in Table 2.5. The nonlinear calibration procedure gave lower reconstruction error than 2D-DLT in the X axis and the resultant direction. Dunn et al. (2012) attributed this to 2D-DLT neglecting to account for lens distortion. These distortions were high as a wide field of view was needed to capture footage of the calibrated plane.

Table 2.5. RMSE of reconstructed test points in the X and Y axes and the resultant (R) direction from Dunn et al. (2012). The 2D-DLT calibration procedure used fifteen control points whereas the nonlinear one used four. n = 60 for inside and n = 64 for outside.

		X (10^{-3} m)	Y (10^{-3} m)	R (10^{-3} m)
Inside	2D-DLT	71.0	39.8	81.4
	Nonlinear	38.1	55.9	67.7
Outside	2D-DLT	105.8	71.6	127.8
	Nonlinear	34.9	69.7	78.0

Advantages

The nonlinear procedure has the following advantages:

- Freely positioned camera. The camera can be located wherever it can capture footage of the calibrated plane.
- Inbuilt support for lens distortion and camera imperfections. Although a zero distortion and perfectly constructed camera is theoretically possible, in practise, most cameras are imperfect and have some distortion (Bradski and Kaehler 2008). The nonlinear calibration procedure can account for this whereas linear-based procedures, e.g. linear scaling and 2D-DLT, may not be able to. This allows for wide field of views, where lens distortion can be large (as shown by Dunn et al. 2012).
- Single camera. A wide field of view makes it more likely that a single camera could be used to obtain footage of a swimming pool. This removes the logistical problems associated with multiple cameras, which are identified in the **Optical axis** section.
- Fine control. Any of the intrinsic parameters or distortion coefficients can be calculated or fixed. In this way, the calibration model need be no more complex than needed, which may help avoid numerical instability (Zhang 2000). In addition, the equations that describe the lens distortion are

interchangeable. For example, Bouguet (2013) provides equations for fisheye and non-fisheye lenses. This means that the equation can be matched with the type of external lens, if any, that was used.

- Inbuilt support for handling noise. Heikkilä and Silvén (1997) state that images always contain noise. The nonlinear calibration procedure is numerically stable, i.e. it dampens the effect of noise on the intrinsic parameters and distortion coefficients. Zhang (2000) did a series of computer simulations to show this: noise led to a 0.3% error in focal length and less than a 1 pixel error in principal point. Using more images of the calibration object than is strictly necessary when calculating the intrinsic parameters and distortion coefficient gives this.

Disadvantages

The nonlinear procedure has the following disadvantages:

- Complexity. The nonlinear calibration procedure is the most complex of those considered in this document. Images of a calibration object must be captured; no other procedure requires this step. In addition, options for the calibration model (e.g. number of distortion coefficients) must be chosen. Although this offers fine control, as will be shown in Chapter 3, an incorrect choice can have a large and detrimental effect on reconstruction error.
- Mathematically challenging. The mathematics that underpins this procedure is not straightforward. This is in stark contrast to the optical axis and linear scaling procedures and, to some extent, 2D-DLT.
- Large depth of field is required for large control volumes. Focused footage of the calibration object and the objects of interest, e.g. the

swimmers, is required. The calibration object is typically held within a metre or so of the camera whereas the swimmers are tens of metres away. Such a depth of field requires a narrow aperture, which can lead to underexposed footage. To ensure a good quality calibration, exposure time can be increased when capturing footage of the calibration object. Then, the exposure time can be reduced to prevent blurring in the race footage.

2.3.8 Conclusion

This section reviewed methods that have or could be used in swimming competition performance analysis and found:

- All of the existing video-based competition performance analysis methods have used either a multi-camera system or a single panning camera.
- Neither of these is ideal. A multi-camera system can cause logistical problems and calibration for a panning camera is time-consuming.
- The nonlinear calibration procedure was the only one with inbuilt support for a lens that could achieve a wide field of view. This may enable a single fixed camera to obtain the footage needed to analyse swimmers in the clean swimming phase. This calibration procedure has not previously been used in swimming competition performance analysis. So, its performance in this application would need assessing.

2.4 Automated swimmer digitisation

2.4.1 Introduction

Section 2.2 identified that competition performance analysis uses manual digitisation of a race’s video footage to split the race into phases and measure a swimmer’s performance in each phase. In this section, ways of automatically doing this are reviewed. Such automated methods use digital image processing techniques. Two commonly used techniques are differencing and thresholding; these are typically used in combination to separate a foreground object from those in the background. Differencing subtracts an image of the background from an image that contains the background and a foreground object. Thresholding classifies as background each pixel that is less than a certain value, which is called the threshold. The other pixels are part of a foreground object. The output of these two techniques is usually an image in which the background is black and foreground is white. An example is in Figure 2.13.

Figure 2.13. Example of thresholding to classify background and foreground pixels (Russ 2011, p402).

No published studies on automated swimming competition analysis could be found. One study that described an automated and video-based system for measuring swimmer lap time in training was found (Pogalin et al. 2007). A small body of work on automated and video-based drowning prevention systems was also found (Eng at al. 2003, Lu and Tan 2004, Wang et al. 2004, and Chan 2013). Arguably the most relevant study in this area remained unpublished. The Centre for Sports Engineering Research at Sheffield Hallam University developed a prototype stroke detection system, called iSwim (Driscoll and Kelley 2013). All these studies are reviewed in this section.

In general, these studies shared a common approach to detecting swimmers in video footage. This approach is called motion-based and assumes that there are two types of object in a scene: a static background and one or more moving foreground objects. In competition performance analysis the swimmers would be the foreground objects and everything else, e.g. the water, lane ropes and pool markings, would be the background objects.

An approach such as this typically involves the following steps:

1. Build models of the background and foreground objects.
2. Use the models to classify the pixels in an image as belonging to either a background or foreground object.
3. Identify the swimmers in the set of foreground objects and track each swimmer from frame-to-frame.
4. Update the models to include any new knowledge about the background and foreground objects gained from steps 2 and 3.

How the studies from the literature addressed the four steps listed above is discussed in the following subsections.

2.4.2 iSwim

This is a prototype of an automated and video-based stroke detection system, which was developed by Driscoll and Kelley (2013) at the Centre for Sports Engineering Research at Sheffield Hallam University. It was targeted at competition performance analysis. In this regard it may be unique: another study on automated and video-based competition performance analysis was not found.

iSwim used video footage from a single fixed camera. An example of the footage used is in Figure 2.14.

Figure 2.14. iSwim used video footage from a single fixed camera.

Strokes were detected using the steps listed below. These are then discussed in more detail in the following subsections.

1. Select a background image, *B*.

2. Construct five profile lines per lane.
3. Read RGB_r for each pixel on each profile line in B, where RGB_r is the red channel in the RGB colour space.
4. Iterate through each image, f_i, in the race footage and:
    4.1. Read RGB_r for each pixel on each profile line in f_i.
    4.2. Calculate the absolute difference between the corresponding values in the RGB_r profile lines for f_i and B.
    4.3. Suppress any small differences in RGB_r.
    4.4. Create a mean profile line.
    4.5. Identify the swimmer's leading edge on the mean profile line.
    4.6. Generate a signal from the rate of change of the leading edge of the swimmer.
    4.7. Process the signal to split the tracking data into laps.
    4.8. Further process the signal for each lap to detect when the swimmer performed a stroke.

Models of background and foreground objects

The background model was a single image when the pool was empty. Typically, this image was from just before the start of the race. An analysis of the colours in such images showed that the background was dominated by RGB_b and RGB_g. In the presence of a swimmer these values changed little yet the amount of RGB_r increased. This gave the swimmer model, i.e. a clear increase in RGB_r above the background value. The RGB_r, RGB_g and RGB_b values along a line in the centre of a lane are shown in Figure 2.15.

Figure 2.15. RGB_r, RGB_g and RGB_b values along a line in the centre of a lane (Driscoll and Kelley 2013, p14).

A key innovation in iSwim was the use of profile lines. A profile line was a line in a swimming lane, which was one pixel wide and aligned with the swimming direction. Five such lines were positioned in the centre of each lane. They had two purposes: a) reduce the computational burden by decreasing the number of pixels that had to be stored and processed and b) omit the lane ropes and pixels that were adjacent to them. It is likely that these pixels were excluded to prevent them being (erroneously) classified as foreground objects, i.e. swimmers. Five profile lines superimposed on a lane is shown in Figure 2.16.

Figure 2.16. The position of five profile lines superimposed on a lane (Driscoll and Kelley 2013, p15).

Classification of pixels

iSwim used a differencing approach to classify the pixels on the profile lines. First, the absolute difference between the RGB_r values of corresponding pixels in the current and background image were calculated. Secondly, small differences in RGB_r were suppressed: any difference in the lower 98th percentile was set to zero. Thirdly, the five profile lines were collapsed into one; this line was the mean of the five it was constructed from. Finally, another suppression of small values was done: any pixel on the mean profile line whose RGB_r value was in the lower 50th percentile was classified as being part of the background and the rest were said to belong to the foreground objects. As there was only one swimmer per lane, all foreground pixels in a lane were assumed to belong to that swimmer.

Tracking of swimmers

In iSwim, the swimmer's position was determined by finding the position of the first and last foreground pixel on the mean profile line. This process is shown in Figure 2.17. The rate of change of the first and last positions was used to split the race into laps, to identify the clean swimming phase in each lap, and to detect when the swimmer performed a stroke in the clean swimming phases.

Figure 2.17. The first and last position of a foreground pixel on the mean profile line (Driscoll and Kelley (2013, p17).

Updating models of background and foreground objects

The background object model was not updated: the same background image was used when processing all images in the race footage. Similarly, the foreground object model was not updated, but it did adapt to the RGB_r values that were in the current and background images: the swimmer was made up of the pixels that had the highest RGB_r values.

Advantages

The iSwim system has the following advantages:

- Single-camera system. The video footage was captured by a single freely positioned camera.
- Simplicity. The background model was a single image and the swimmer was identified by a simple process involving two fixed percentiles. Also, swimmer tracking was straightforward.
- Profile lines. These reduced the number of pixels that had to be read from hard disk, stored in memory, and processed. In addition, they could be positioned to exclude problematic regions of the image (i.e. the lane ropes).
- Low computational burden. First, the RGB colour space was used; this is the native colour space in many development tools and hence conversion to another colour space was not required. Secondly, as discussed above, the use of profile lines had a lower computational burden than if all pixels in each lane were used. Finally, iSwim used simple models that were not updated.

Disadvantages

The iSwim system has the following disadvantages:

- Modelling of non-static background. iSwim made no attempt to model changes in the background. Instead, some of the background motion was omitted by careful positioning of the profile lines. Such positioning may not be suitable for tracking swimmers who are adjacent to the lane ropes.

2.4.3 Lap time measurement system

This is an automated and video-based lap time measurement system, which was developed by Pogalin et al. (2007) in conjunction with the Dutch Olympic Committee. It was targeted at the competitive swimming training environment. Three pool-side, fixed view cameras, with overlapping views, were used to film the swimming that took place in three lanes of a 50 m swimming pool. The lanes had one to four swimmers in them. The camera positions and example footage is shown in Figure 2.18.

Figure 2.18. Camera positions and example footage for the lap time measurement system (Pogalin et al, 2007, p6 and p7).

The footage was distorted. This was undistorted using the nonlinear calibration procedure as implemented by Bouguet (2013). Point reconstruction was achieved with the linear scaling algorithm; control points on the pool deck and lane ropes were used to calculate the scaling coefficient (in Equation 2.1).

Lap times were measured by iterating through each image, f_i, in the training footage and applying the process detailed below:

1. Read RGB for each pixel in each lane in f_i.
2. Convert the RGB to the Cr channel of the YCbCr colour space.
3. Update background image, B_i, using a weighted average scheme.
4. Calculate the difference between the Cr values in f_i and B_i.
5. Threshold the resulting differences.
6. Associate connected regions of foreground pixels in f_i with those from previous frames.

Models of background and foreground objects

Pogalin et al. (2007) noted that the pool was largely blue and the swimmers were mainly red. This was a similar finding to that reported by Driscoll and Kelley (2013). In addition, they found that the difference between pool and swimmer was enhanced by converting from the RGB to the YCbCr colour space. Only the Cr channel of this colour space was subsequently used.

The background model in frame i, i.e. B_i, was a weighted average of the previous background model and the current image, f_i:

$$B_i = (1-\alpha)B_{i-i} + \alpha f_i \qquad 2.24$$

where α is the background model update rate. In this way, the background model was continuously updated.

The foreground objects, i.e. the swimmers, were not directly modelled. Instead, a pixel not considered to be part of a background object was a swimmer. This was similar to the approach used by iSwim.

Classification of pixels

Pogalin et al. (2007) classified pixels by way of background subtraction. The difference between the Cr values of corresponding pixels in the current and background image were calculated. The difference values were thresholded and foreground pixels were formed into connected regions. If such regions contained a certain number of pixels and a fraction of these had difference values above an upper threshold, then all the pixels in that region were classified as being a foreground object, i.e. a swimmer. All other pixels were classified as background.

Tracking of swimmers

Pogalin et al. (2007) tracked swimmers between frames, cameras and laps. If a connected region of foreground pixels overlapped one from the previous frame, then these two regions were said to be the same swimmer. Swimmers were tracked across cameras by selecting the regions that had the “best degree of overlap” (Pogalin et al. 2007, p9) and were moving in the same direction. Tracking swimmers between laps proved to be problematic as they were either underwater or static (and therefore less red than a moving swimmer). Methods for handling this and other challenges, such as the merging and splitting of swimmers and incomplete tracking, were presented. These included manual interventions.

Advantages

The lap time measurement system has the following advantages:

- Simplicity. The background model was constructed and updated via a simple equation and the foreground objects were classified through a straightforward process involving two fixed thresholds.
- Regular background model updates. This accounted for the highly dynamic nature of the swimming pool environment.
- Swimmers as connected regions. It was recognised that swimmers must be a connected region of foreground pixels that was a certain size. This was used to inform the classification of pixels. As a by-product, small non-static background objects (e.g. ripples, reflections) that were in the same part of the colour space as the swimmers were rejected.

Disadvantages

The lap time measurement system has the following disadvantages:

- Multiple cameras. The system was time-consuming to setup: each camera had to be positioned and orientated so that it overlapped with the adjacent one and then calibrated. The authors identified that such a system would ideally have fewer cameras.
- Modelling of non-static background. Reflections and ripples were not modelled and occasionally prevented identification of a swimmer or led to background objects being classified as a swimmer. Examples of this are shown in Figure 2.19.
- Above water and motion-based approach for tracking underwater or static foreground objects. Swimmers at the end of a lap proved hard to track.

a) **b)**

Figure 2.19. Reflections and ripples a) preventing swimmer identification and b) incorrectly identified as a swimmer (Pogalin et al. 2007, p12).

2.4.4 Drowning prevention systems

Systems aimed at the prevention of drowning in swimming pools are described in this subsection. This involves tracking swimmers in video footage from above water fixed cameras and identifying the early signs of a water crisis. Four studies illustrate the progress in this area. These are discussed in more detail in the following subsections.

- Eng at al. (2003). Introduced block-based models for background and foreground objects.
- Lu and Tan (2004). Used a Kalman filter to track swimmers. This provided a local search window for subsequent swimmer identification. Devised a scheme to remove the splash created by the swimmer.
- Wang et al. (2004). Introduced pixel motion frequency, which characterised the temporal variation in a pixel’s colour. This was used to classify and filter pixels and hence reduce noise and specular reflections.

- Chan (2013). Developed the pixel motion concept to classify the background and different parts of the swimmer. This provided a local search window for swimmer identification.

These studies share a common approach to identifying and tracking swimmers:

1. Take a sequence of images:
    1.1. Learn background and foreground models.
    1.2. Classify each pixel using its motion frequency, if used.
2. Iterate through each image, f_i, in the sequence and:
    2.1. Calculate the difference between each pixel in f_i and the background and foreground models. A local search window may be used.
    2.2. Use the difference to classify each pixel as belonging to either a background or foreground object.
    2.3. Identify the swimmers and associate them with those from previous tracking.
    2.4. Update the background and foreground models.

Models of background and foreground objects

The drowning prevention systems used a variety of colour spaces. Eng et al. (2003) found that the CIELa*b* provided better segmentation, i.e. larger differences between background and foreground objects, than other spaces. Lu and Tan (2004) used HSV and Chan (2013) used RGB.

All studies constructed a background model by clustering in their chosen colour space. This approach was motivated by noting that a swimming pool contained only a few background objects that formed tight clusters in the colour space. Example objects are water, pool floor, and pool floor markings. A background model that consisted of two clusters in HSV colour space is shown in Figure 2.20.

Figure 2.20. A background model consisting of two clusters in the HSV colour space (Lu and Tan 2004, p162 and p163).

Lu and Tan (2004) formed clusters for the whole pool whereas the others used a block-based approach. This split the pool into a grid of non-overlapping blocks, e.g. Chan (2013) split the pool into a grid of 30 by 32 pixel blocks and formed clusters for each block.

Eng et al. (2003) and Wang et al. (2004) used k-means clustering to find up to three cluster centroids per block. Lu and Tan (2004) used a Gaussian Mixed Model: this described the pool as a set of distributions, which each had its own mean and standard deviation. Chan (2013) combined elements of both these methods, i.e. k-means to find clusters, which were then characterised by the mean and standard deviation of their Gaussian distribution.

All the studies recognized that the background objects were subject to temporal variations. These variations were caused by water movement, reflections, shadows, splashes and ripples. Examples of the effect of these on a pool floor marking are shown in Figure 2.21. The consensus was that such movements had to be accounted for when building a background model.

Figure 2.21. A pool floor background pixel affected by water movement and ripple in a short sequence of images (Lu and Tan 2004, p162).

One approach, used by Lu and Tan (2004), was to build a background model from images when the background was not moving, i.e. an empty and still swimming pool. The other studies constructed a clean background image (*B*) from a sequence of images, e.g. Chan (2013) used sequences of five or twelve images. A temporal median filter was used to construct *B*. This filter sets a pixel in *B*, i.e. *B(U, V)*, to be the colour that was the closest to the corresponding pixels in the image sequence using:

$$B(U,V) = \left\{ f_j(U,V) \in f(U,V) \middle| \min_{f_j(U,V)} \sum_{i=1}^{N} \left| f_i(U,V) - f_j(U,V) \right| \right\} \qquad 2.25$$

where $f_n(U,V)$ is the colour of pixel (U,V) in image n. This is Equation 1 from Eng et al. (2003). A clean background image constructed using a temporal median filter is shown in Figure 2.22.

Figure 2.22. A clean background image constructed with a temporal median filter from a sequence of images that contained a swimmer (Chan 2013, p82 and p85).

Two approaches to constructing a model of a foreground object, i.e. a swimmer, were reported. Eng et al. (2003) and Lu and Tan (2004) used the same approach they used to model the background, i.e. clustering in colour space. In contrast, Wang et al. (2004) and Chan (2013) did not directly model the swimmers. Instead, a pixel not considered to be part of a background object was a swimmer.

The modelling described so far was colour-based. Wang et al. (2004) and Chan (2013) did an additional modelling step, which characterised how a pixel's colour changed over a sequence of images. This was called pixel motion frequency and was used to identify certain classes of background or foreground objects.

Wang et al. (2004) classified pixels as low, medium or high frequency. The background was low and noise was high. The medium class contained a mixture of non-static background objects and the foreground objects. These classes were used to determine which filter to apply to a pixel. Low frequency pixels were not filtered, noise was supressed by applying a spatial mean filter, and a temporal median filter reduced specular reflections for the medium frequency class.

Chan (2012) used pixel motion frequency to define a search window that identified the most likely location of the swimmers in the image. A periodogram was used to characterise the frequencies; this is a histogram of frequency for a certain class of object. A periodogram for a swimmer's head is shown in Figure 2.23. Classes were no motion, random motion, water ripple and swimmer's head, upper trunk and hand. Pixels were classified by comparing the amplitude, frequency and mode of its distribution to the learnt periodograms.

Figure 2.23. Periodogram for a point on swimmer's head calculated over thirty frames (Chan 2013, p79).

Classification of pixels

The logic used by Eng et al. (2003) to classify pixels as belonging to either a background or foreground object is shown in Figure 2.24.

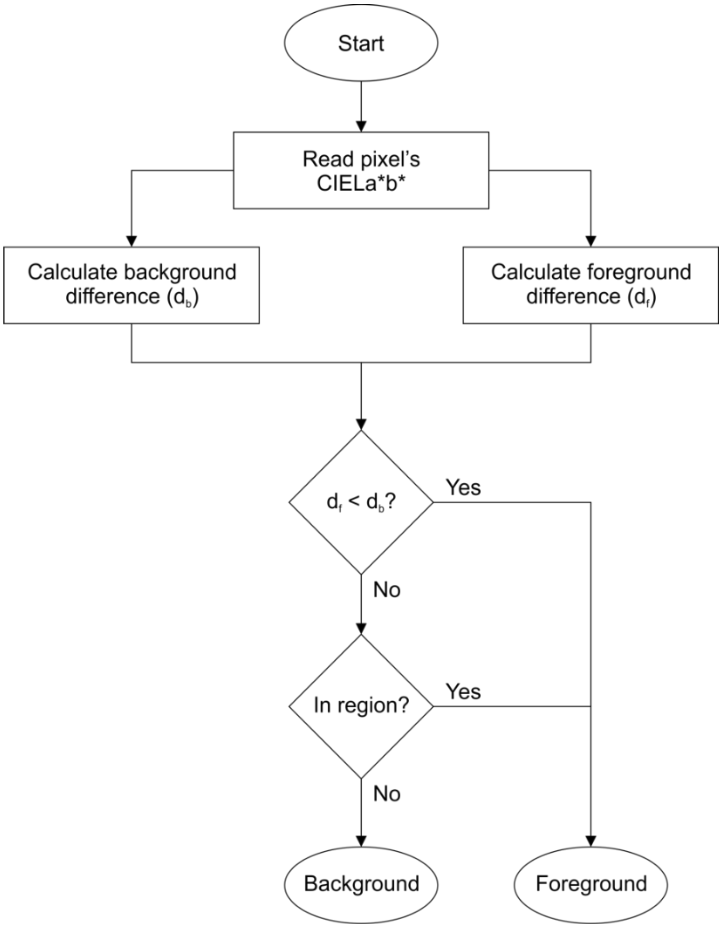


Figure 2.24. Pixel classification logic for Eng et al. (2003).

The distances between a pixel and the block-based background and foreground models (i.e. d_b and d_f in Figure 2.24) were calculated for an eight-connected region of blocks. The block that enclosed the pixel and the eight blocks that surrounded it were used. The minimum distance between the pixel's colour and the centroids in these nine blocks was found; this gave d_b and d_f.

Pixels for which d_f was less than d_b were classified as belonging to a foreground object. Other pixels were subjected to a thresholding scheme. All pixels whose d_b value was above a lower threshold were considered to be candidate foreground pixels. These were formed into connected regions. If such regions contained a certain fraction of pixels whose d_b value was above an upper threshold, then all the pixels in that region were classified a part of a foreground object. All other pixels were classified as background.

Lu and Tan (2004) used similar logic to Eng et al. (2003). This is shown in Figure 2.25.

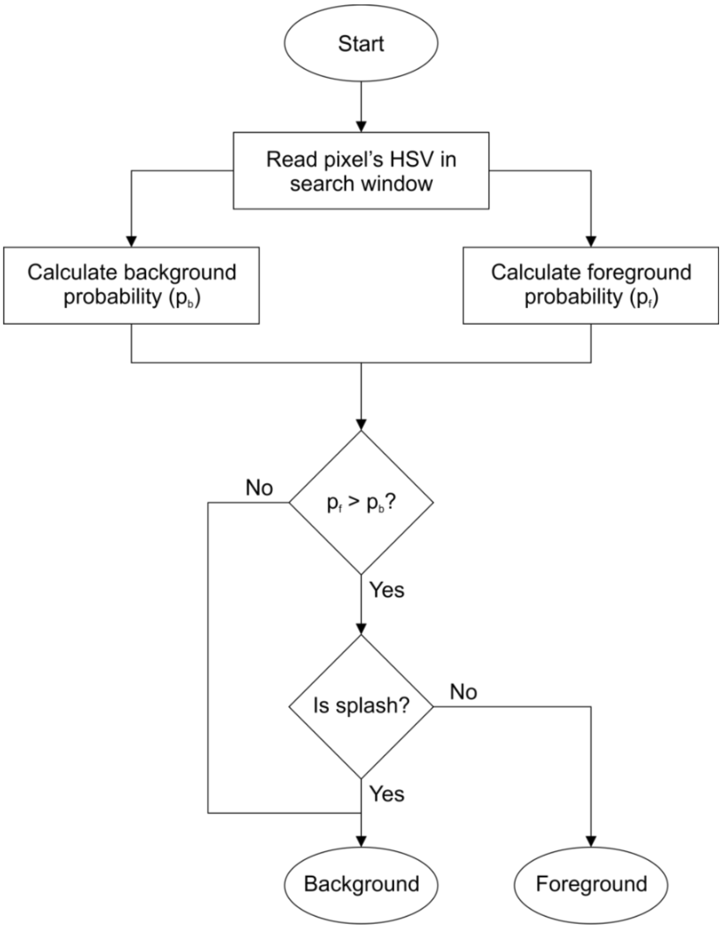


Figure 2.25. Pixel classification logic for Lu and Tan (2004).

The probability that a pixel was part of a background or foreground object was calculated. This gave p_b and p_f (in Figure 2.25). If p_f was greater than p_b, then the pixel was initially classified as belonging to a foreground object. All other pixels were classified as being part of the background. The candidate foreground pixels were subjected to a further test. This used the pixel's HSV_V, i.e. the intensity of the pixel. If the intensity was less than 0.9 of the average intensity of the background, then the pixel was classified as a foreground object, i.e. a swimmer. This was based on the observation that a swimmer’s intensity was less than that of the water and the splashes created by the swimmer’s presence. An illustration of this is in Figure 2.26.

Figure 2.26. HSV$_V$ of pixels in region around a swimmer (Lu and Tan 2004, p165).

Wang et al. (2004) used a similar classification scheme to Eng et al. (2003). Key differences were the pre-processing of images using pixel motion frequency and no swimmer models. This gave the logic shown in Figure 2.27, i.e. after pixel motion frequency filtering, the pixels were subjected to the thresholding scheme used by Eng et al. (2003).

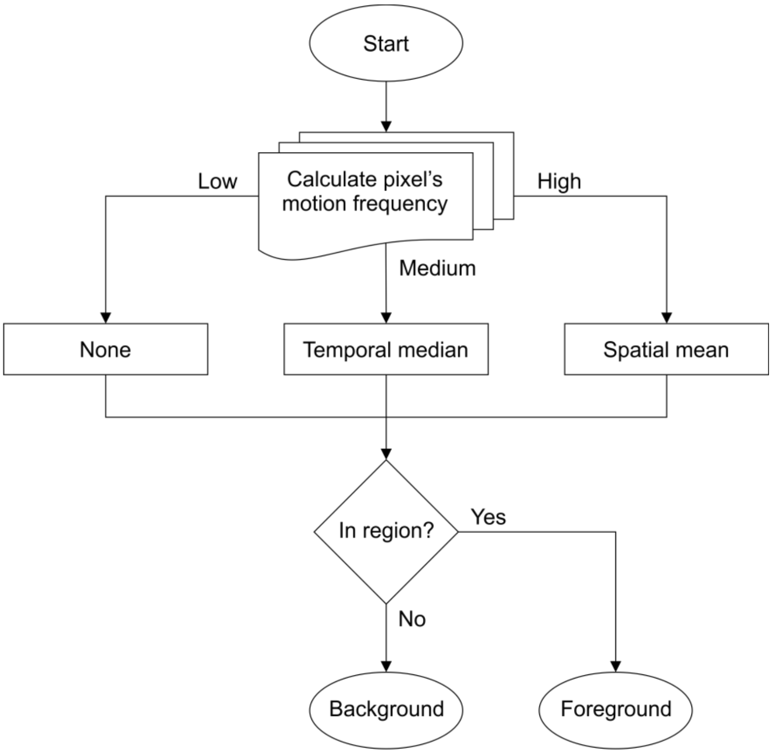


Figure 2.27. Pixel classification logic for Wang et al. (2004).

Chan (2013) used pixel motion frequency to perform an initial classification of pixels. This produced a motion map in which pixels most likely to be part of a swimmer were set to white. An example motion map is shown in Figure 2.28.

Figure 2.28. Motion map (Chan 2013, p81 and p83). The pixels most likely to belong to the swimmer were set to white.

The difference between the RGB colours of the pixels in the motion map's white region and the background model were calculated. A 5 by 5 grid of blocks in the background model, centred on the one that enclosed the pixel, was used. Any pixel for which this difference was less than a threshold was classified as part of the background and the other pixels were said to belong to a foreground object.

Chan's (2013) process for classifying pixels is shown in Figure 2.29.

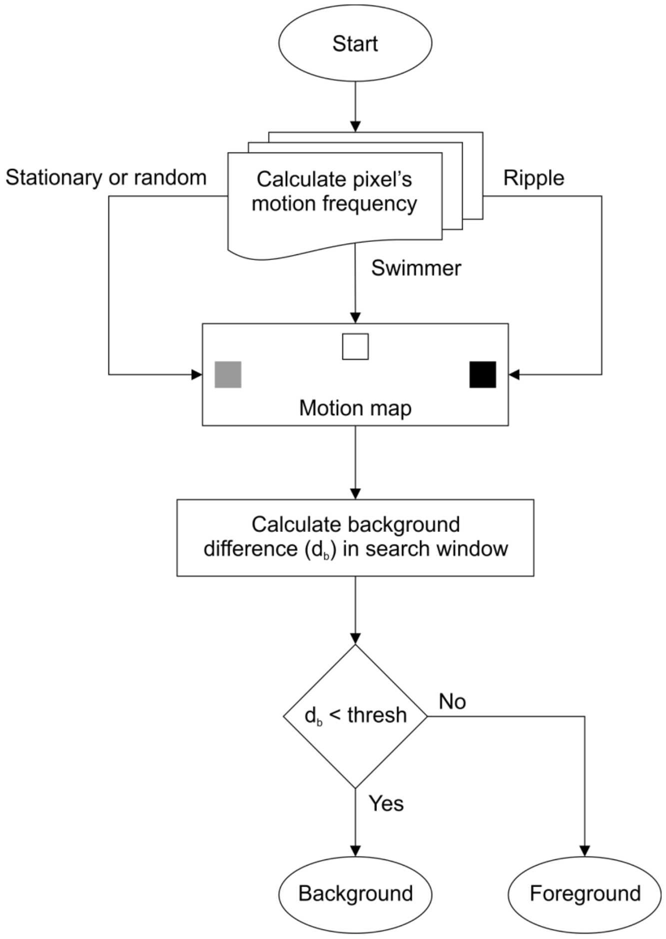


Figure 2.29. Pixel classification logic for Chan (2013).

Tracking of swimmers

The first step in tracking swimmers was to label the connected regions of foreground pixels. Then, two methods of identifying the same swimmer in two consecutive frames were used. Eng et al. (2003) and Wang et al. (2004) associated the swimmers that had the minimum Mahalanobis distance between them. Lu and Tan (2004) used a Kalman filter, which characterised a swimmer as a current position and an estimated inter-frame displacement. The swimmer that was closest to the predicted position was said to be the same swimmer. Chan (2013) did not track swimmers.

Updating models of background and foreground objects

The background models were periodically updated. Chan (2013) provided an explanation of how this was done: the model was updated, if needed, every fifth frame; this involved a weighted adjustment to the mean and standard deviation of the distributions in each block using a weighted average method similar to that presented in Equation 2.24.

Advantages

The drowning prevention systems had the following advantages:

- Advanced models of the background and foreground objects. Clustering captured the object's main characteristics. This, with the use of the smallest distance between a pixel and the clusters that surrounded it, supressed small variations in the images. In effect, the dominant objects were identified, modelled and searched for. Pixel motion frequency was used to remove known problems, e.g. noise and specular reflections. In addition, it was used to define a search window where the swimmer was most likely to be; this removed most non-swimmer pixels at an early stage. Altogether, these advanced models increased the likelihood of a correct classification of pixels.
- Use of clean background images. The background models were constructed from an image that averaged out temporal variations. In general, a value that is the average of several measurements is more likely to represent the true value than is a single measurement.

Therefore, it can be expected that this approach better characterises the background than does a single image.

- Regular model updates. This accounted for the highly dynamic nature of the swimming pool environment.
- Reduced memory requirement. Representing the background and foreground objects as clusters in colour space required less memory than would have been needed to store the colour of all pixels.
- Swimmers as connected regions. It was recognised that swimmers must be a connected region of foreground pixels that was a certain size. This was used to inform the classification of pixels.

Disadvantages

The drowning prevention systems had the following disadvantages:

- Complexity. The construction, use and update of the advanced models of the background and foreground objects are not easy to understand or implement. As a result, specialist knowledge and software, e.g. MATLAB with the statistics toolbox, would be required.
- Large execution time. This complexity had another cost: high execution time. Chan et al. (2013) reported the execution time for their system, which ran in MATLAB on a computer with a 2.1 GHz CPU and 1 GB of RAM. It took 21 minutes and 6 seconds to process 60 frames. In contrast, Eng et al. (2003) ran their simpler system in real time at a frame rate of 4 Hz.

2.4.5 Conclusions

Section 2.2 identified that competition performance analysis used manual digitisation. This section reviewed ways of automatically doing this and found:

- A key challenge is how to use digital image processing techniques to identify swimmers in video footage where the background is not static.
- A regularly updated model of the background is probably needed. Such models can be complex and may have large execution times.
- The camera that captures the footage must have a fixed view. Ideally, a single camera would be used.

2.5 Summary

The clean swimming phase was identified as a key phase in a race. This phase is at least 30 m long in the X axis.

Automated digitisation of the swimmers in the clean swimming phase of a race should be possible. Digital image processing techniques could achieve this. The highly dynamic nature of the swimming pool needs to be modelled.

A single-camera system is preferable to a multi-camera one. In addition, for automated digitisation, the camera must have a fixed view.

The nonlinear calibration procedure is probably the best way of measuring distances for such a system. The accuracy and precision with which it could do this has not been assessed. Figure 2.30 identifies the different errors in reconstructed distances that should be considered.

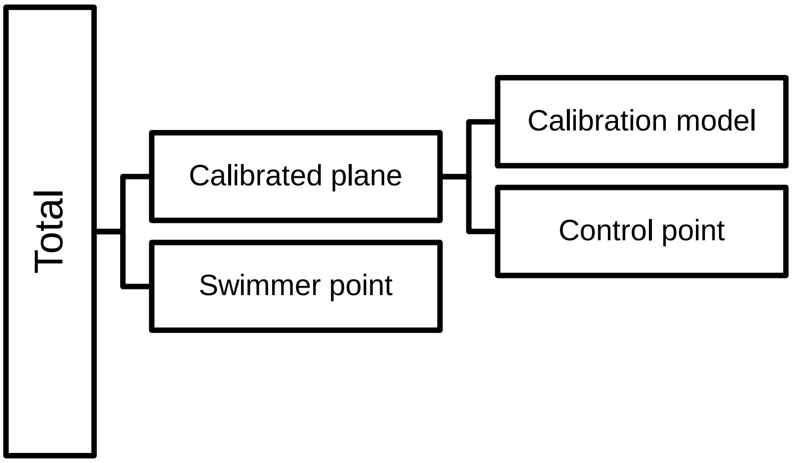


Figure 2.30. Errors in distances reconstructed using the 2D-DLT and nonlinear calibration procedures.

The errors in Figure 2.30 are described below:

- Calibration model. A calibration model is used to convert distances in the video footage to distances in the swimming pool. So, calibration model error is caused by not using the correct relationship between distances in the video footage and the swimming pool.
- Control point. A calibration model is calculated from control points. These provide information about the relationship between distances in the video footage and the swimming pool. The control points must be measured and are therefore subject to error. So, control point error quantifies how uncertainty in the measurement of the control points propagates to error in reconstructed distances.
- Calibrated plane. This is the combination of calibration model and control point errors. It shows how sensitive a calibration model is to uncertainties in the measurement of the control points that is was calculated from.
- Swimmer point. A swimmer point is the point on a swimmer that was digitised. As a measured value, it is subject to error. So, swimmer point

error quantifies how uncertainty in the digitisation of a swimmer propagates to error in reconstructed distances.

- Total. This is the combination of all other errors. It shows how errors in the construction of a calibrated plane and the digitisation of swimmers on that plane propagate to error in reconstructed distances.

2.6 Aim and objectives

2.6.1 Aim

To establish a process to enable the confident, accurate and precise use of a wide field of view for measuring distance and speed in sports analysis.

2.6.2 Objectives

1. To establish a process for determining the accuracy and precision with which distance and speed can be measured from video footage of a sports competition that was captured with a wide field of view.
2. To demonstrate how a wide field of view can be used to measure the performance of elite athletes in a competition.
3. To show how to automate the measurement of an athlete’s speed from video footage of a sports competition that was captured with a wide field of view.

3 Calibration model error

3.1 Introduction

This chapter calculated calibration model error. This was the error that was caused by a calibration model not having the correct relationship between distances in the pixel and world coordinate systems. Figure 3.1 shows its place in the hierarchy of errors; all the other errors in this hierarchy were fixed at zero in this chapter.

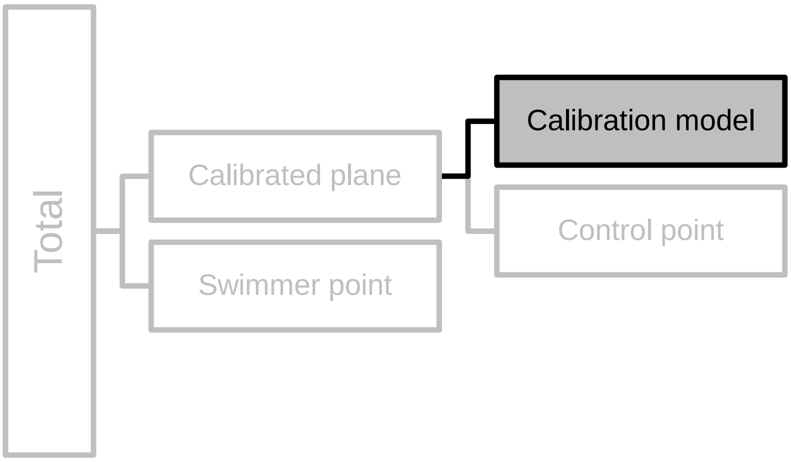


Figure 3.1. Chapter 3 calculated calibration model error. Control point and swimmer point errors were fixed at zero.

Calibration model error was calculated for different viewpoints. A viewpoint encapsulated the camera location, orientation, external lens and settings that were used to capture video footage of a swimming pool. Distinctions between viewpoints were:

- Single- or multi-camera. A multi-camera viewpoint had a view of only part of a swimming pool; this represented the traditional approach to competition performance analysis in which two or more cameras were used (e.g. Arellano et al. 1994, Hellard et al. 2008). A single-camera viewpoint captured footage of the whole swimming pool.

- Perpendicular or non-perpendicular. A perpendicular viewpoint had the optical axis perpendicular to the swimming direction whereas a non-perpendicular one did not.
- Fisheye or non-fisheye. A fisheye viewpoint used a fisheye lens converter to fit the whole swimming pool into the camera's view whereas a non-fisheye one did not.

In addition to different viewpoints, calibration model error was also calculated for a variety of calibration models. The 2D-DLT and nonlinear calibration procedures were used to construct these models.

The benchmark calibration model error was from Veiga et al. (2010). A three camera system was used in this study. The viewpoints were multi-camera, perpendicular and non-fisheye and the 2D-DLT calibration procedure was used. Calibration model error was 46×10^{-3} m. Veiga et al. (2010) used a real swimming pool to calculate calibration model error. Control points were fixed landmarks on the pool deck and test distances were on the floating lane ropes. In contrast, this chapter used a checkerboard scale swimming pool, which is shown in Figure 3.2. Control points and test distance endpoints were checkerboard corners. This approach was chosen as it offered highly accurate and precise world and pixel coordinates for the control points and test distance endpoints, an accessible and controlled environment (especially in terms of lighting) for filming, and the choice of any realistic viewpoint.

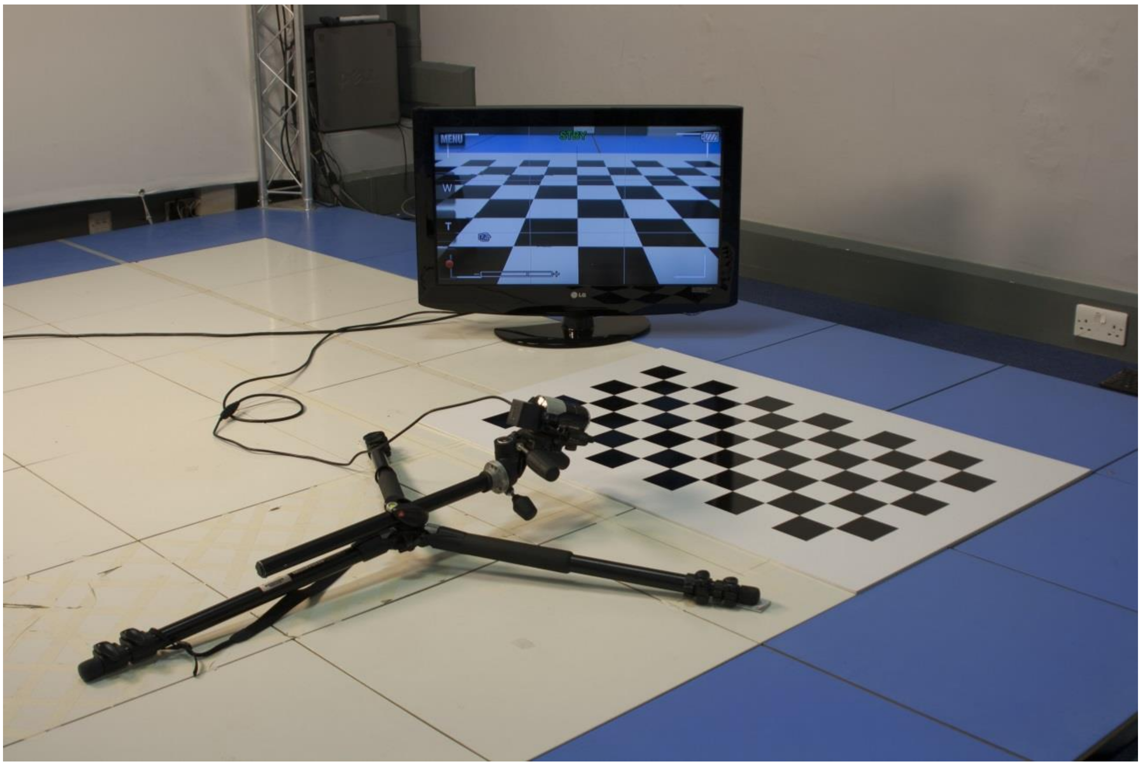


Figure 3.2. Experimental setup used to calculate calibration model error. A checkerboard scale swimming pool was used to calculate this error.

This chapter also calculated a quantity called distortion. This was used to highlight the effect that a fisheye lens converter had on the video footage captured with it. In this way, it gave an indication of what a calibration model had to account for when reconstructing distances. An example of the distortion caused by a fisheye lens converter is shown in Figure 3.3.

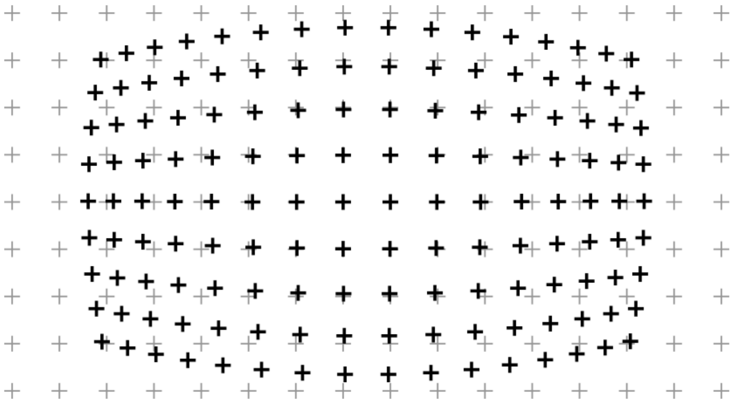


Figure 3.3. Distortion caused by a fisheye lens converter.

3.2 Method

A summary of the method is given below. Each of these steps is described in more detail in the following subsections.

1. A scale 50 m by 25 m swimming pool was created. See the **Scale swimming pool** section.
2. Video footage of this calibrated plane was captured from four single-camera viewpoints and a multi-camera viewpoint. The world and pixel coordinates of four control points and test distance endpoints were determined from this footage. See the **Viewpoints** section.
3. Calibration models were calculated from control point coordinates and, for the nonlinear calibration models, video footage of a planar calibration object. There were twenty one calibration models for the fisheye viewpoints and thirty three for the non-fisheye viewpoints. Each calibration model used either a different calibration procedure (i.e. 2D-DLT or nonlinear) and, for the nonlinear ones, different options when estimating intrinsic parameters and distortion coefficients. See the **Calibration models** section.
4. Each calibration model was used to reconstruct test distances. Calibration model error was calculated. An optimal calibration model was selected for each viewpoint: this had the best compromise between calibration model simplicity and low calibration model error. See the **Reconstruction** section.
5. Distortion was calculated for the optimal calibration model for each viewpoint. See the **Distortion** section.

3.2.1 Scale swimming pool

A black and white checkerboard pattern was printed onto a flat board. The pattern consisted of a twelve by seven grid of 80 x 10^{-3} m squares. A scale factor of 62.5 was used to convert measurements of the checkerboard pattern to measurements of a real swimming pool. So, each square was 5 m and the central ten by five grid represented the water surface of a 50 m by 25 m swimming pool. All measurements in this chapter were scaled by 62.5 to characterise measurements of a real swimming pool.

3.2.2 Viewpoints

Four single-camera viewpoints (V1 to V4) and a multi-camera viewpoint (V5) were chosen. Camera locations for the viewpoints are shown in Figure 3.4. The calibrated plane for V1 to V4 was 50 m in the X axis and 25 m in the Y axis. For V5 it was 20 m in the X axis and 25 m in the Y axis.

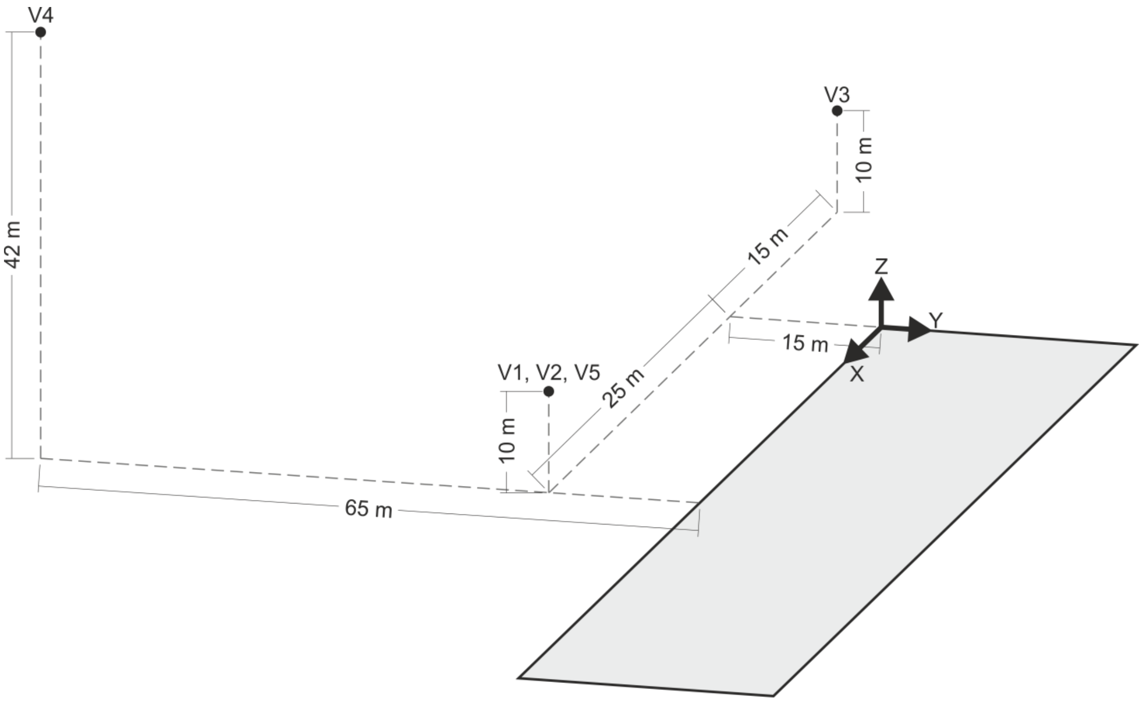


Figure 3.4 Camera locations for viewpoints Vi (i = 1 to 5).

V1, V2, V3 and V5 used camera locations that were based on the dimensions of the spectator seating at the Ponds Forge International Sports Centre. These were representative of viewpoints available at a typical swimming competition venue. V4 was based on a camera location used by British Swimming at the 2012 Olympic Games. This was an atypical viewpoint: the camera was located at the top of a temporary stand that was removed post-Games.

V1 and V2 needed a fisheye lens converter to fit the calibrated plane into the image. To test the effect of different lenses on calibration model error, V1 used a 0.30 multiplier fisheye lens converter and V2 a 0.42 multiplier fisheye lens converter. The camera location for V3 was selected to fit the calibrated plane into the image without the need for a fisheye lens converter. V4, due to the distance between the camera and calibrated plane, did not require a fisheye lens converter. V5 captured an image of the calibrated plane that was 20 m in the X axis; this was the largest length that could be captured from the camera's location without using a fisheye lens converter. In addition, it was representative of the size of calibrated plane used in the literature (e.g. Arellano et al. 1994, Hellard et al. 2008). The viewpoints are summarised in Table 3.1.

Table 3.1. Summary of viewpoints Vi (i = 1 to 5).

	V1	V2	V3	V4	V5
Camera X (m)	25	25	-15	25	25
Camera Y (m)	-15	-15	-15	-65	-15
Camera Z (m)	10	10	10	42	10
Calibrated plane X (m)	50	50	50	50	20
Perpendicular viewpoint?	Yes	Yes	No	Yes	Yes
Fisheye viewpoint?	Yes	Yes	No	No	No

A Sony HDR-PJ260VE (Sony 2015) camcorder was used to capture Full High-Definition (FHD) video footage of the calibrated plane and a planar calibration object. First, the camcorder was positioned and orientated for V1. Next, it was zoomed and focused. Then, the camera’s settings were locked and footage of the calibrated plane and a planar calibration object was captured. This process was then repeated for V2 to V5 to give footage of the calibrated plane and calibration object for each viewpoint. Example images from the footage of the calibrated plane are shown in Figure 3.5.

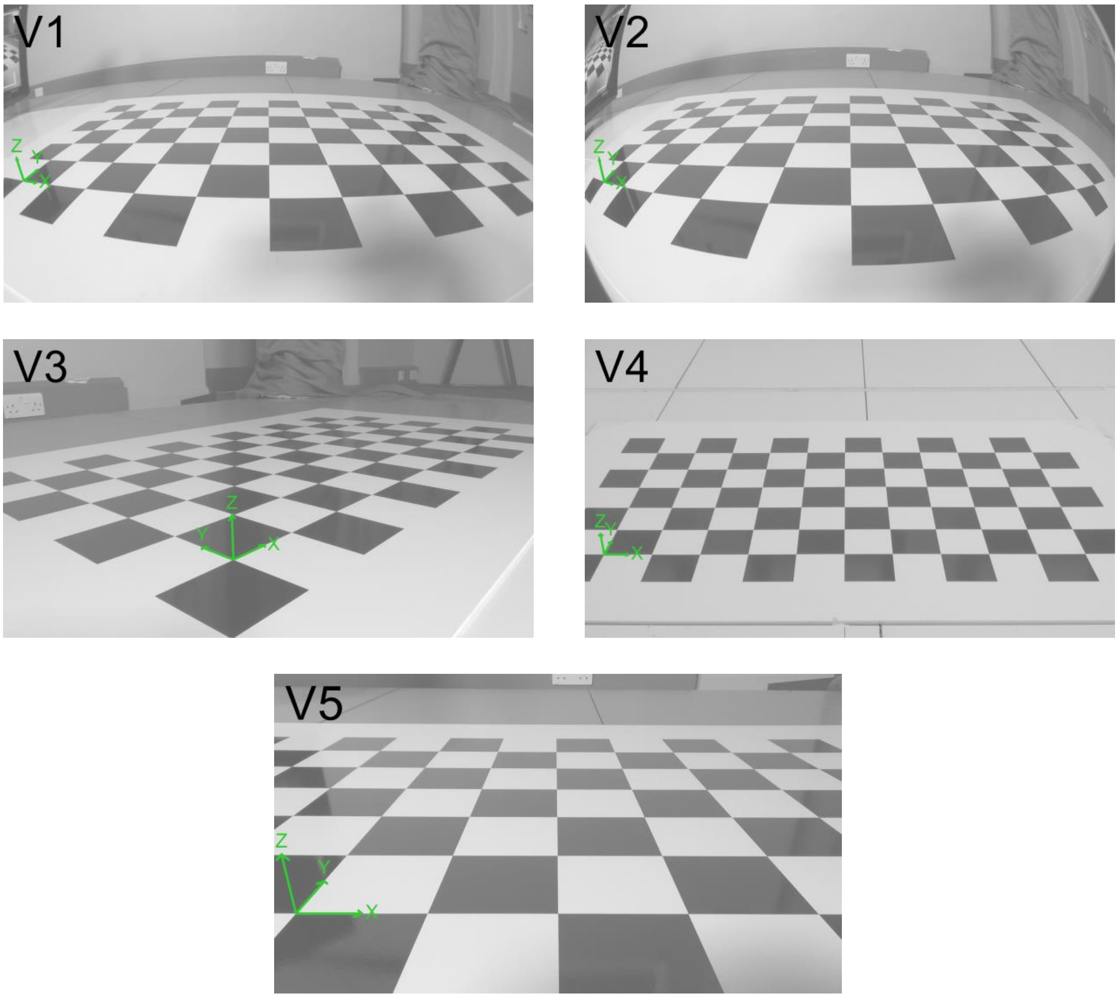


Figure 3.5. Images of the calibrated plane taken from the four single-camera viewpoints (V1 to V4) and multi-camera viewpoint (V5).

The four corners of the calibrated plane were used as control points, Ci (i = 1 to 4). This was the smallest number of control points that enclosed the calibrated plane and the minimum required to calculate a calibration model. The remaining corners were used to construct test distances. Each test distance was between consecutive corners in the X axis and had a ground truth distance of 5 m. V1 to V4 had fifty six test distances and V5 had twenty. The control points and test distances are shown in Figure 3.6.

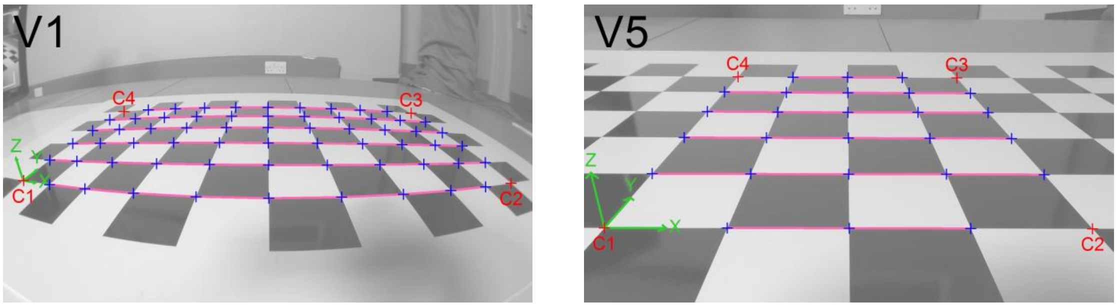


Figure 3.6. Control points, Ci (i = 1 to 4), in red, test distances in pink, and test distance endpoints in blue.

The world coordinates of the control points and test distance endpoints were known from the checkerboard geometry and the scale. The pixel coordinates were calculated by a semi-automated and sub-pixel accurate MATLAB (MathWorks 2014) function provided by Bouguet (2013). This function required the user to provide an initial estimate of the control points' pixel coordinates and the number of grid squares in the calibrated plane. The function then used digital image processing techniques to automatically determine the sub-pixel accurate pixel coordinates of the control points and test distance endpoints.

3.2.3 Calibration models

Different calibration models were used to reconstruct the test distances for each viewpoint. The first was calculated using the 2D-DLT calibration procedure. The others used the nonlinear calibration procedure. In total, twenty nonlinear calibration models were calculated for the fisheye viewpoints and thirty two for the non-fisheye ones. The nonlinear distortion coefficients used matched the viewpoint, i.e. the fisheye ones used k_i (i = 6 to 9) and the non-fisheye ones used k_i (i = 1 to 5). Each nonlinear calibration model varied which intrinsic parameters and distortion coefficients were calculated: a fixed radial distortion coefficient was fixed at zero, not calculating tangential distortion fixed k_4 and k_5 at zero, the fixed principal point was at the image centre, and the fixed aspect ratio was one. The different options used when calculating nonlinear calibration models are summarised in Table 3.2. The term |*k*| was used to represent the number of radial distortion coefficients that were calculated.

Table 3.2. The nonlinear calibration models varied which intrinsic parameters and distortion coefficients were calculated.

	V1, V2	V3, V4, V5
|*k*|	0, 1, 2, 3 or 4	0, 1, 2 or 3
Calculate tangential?	N/A	Yes or No
Calculate principal point?	Yes or No	Yes or No
Calculate aspect ratio?	Yes or No	Yes or No
Count	20	32

MATLAB scripts from the International Society of Biomechanics (2014), Bouguet (2014) and Dunn et al. (2012) were used to do calculate the calibration models. Reprojection error (as described in Section 2.3.7) was calculated for each nonlinear calibration model.

3.2.4 Reconstruction

Each calibration model was used to reconstruct the test distances. The reconstruction was done using Equations 2.5 and 2.6 for the 2D-DLT calibration models and 2.18 to 2.23 for the nonlinear ones. These equations were implemented in MATLAB scripts from the International Society of Biomechanics (2014), Bouguet (2014) and Dunn et al. (2012).

Calibration model error was calculated for each calibration model. It was calculated as the RMSE of n reconstructed distances:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(d_i' - d)^2} \qquad 3.1$$

where d_i' was a reconstructed distance and d was the ground truth distance. The error in a single reconstructed distance, i.e. $d_i' - d$, was called reconstructed distance error, or RDE.

An optimal calibration model was selected for each viewpoint. This was the one that was the best compromise between low calibration model error and simplicity. A 2D-DLT calibration model was deemed to be simpler than a nonlinear one and when comparing different nonlinear calibration models the one with the fewest estimated distortion coefficients and intrinsic parameters was considered the simplest.

3.2.5 Distortion

Distortion was calculated for the optimal calibration models using the following steps:

1. Defined the pixel coordinates of undistorted points, which were at the centre of 120 pixel squares in a 1920 pixel wide and 1080 pixel high image. This gave a total of 144 points.
2. Transformed the points' pixel coordinates to normalised image coordinates using Equation 2.18.
3. Added distortion using Equations 2.11 to 2.13 for non-fisheye viewpoints and Equations 2.14 to 2.16 for fisheye viewpoints.

4. Transformed to pixel coordinates using Equation 2.17.
5. Calculated distortion as the Euclidean distance between corresponding points from step 1 and step 4.

These steps were implemented in a MATLAB script adapted from Bouguet (2013).

3.3 Results

Calibration model error for all viewpoints is in Appendix 15.1.

3.3.1 V1

This was one of the perpendicular and fisheye viewpoints. It used a 0.30 multiplier fisheye lens converter.

2D-DLT gave the highest calibration model error (1018.3 x 10^{-3} m) followed by the nonlinear calibration models with a |*k*| of zero (96.8 x 10^{-3} m to 144.7 x 10^{-3} m). Once these models were discounted then calculating the principal point always led to the lowest calibration model errors (16.0 x 10^{-3} m to 16.6 x 10^{-3} m) and a |*k*| of one resulted in the smallest calibration model error. Calculating aspect ratio did not reduce calibration model error. So, the optimal calibration model was nonlinear with a |*k*| of one, a calculated principal point and a fixed aspect ratio. Calibration model error for this calibration model was 16.0 x 10^{-3} m.

3.3.2 V2

This was one of the perpendicular and fisheye viewpoints. It used a 0.42 multiplier fisheye lens converter.

2D-DLT gave the highest calibration model error (1353.5 x 10^{-3} m) followed by the nonlinear calibration models with a |*k*| of zero and one (83.6 x 10^{-3} m to

627.0×10^{-3} m). Once these models were discounted then fixing the principal point at the image centre always led to the lowest calibration model error (20.4×10^{-3} m to 32.2×10^{-3} m). A $|k|$ of three or four resulted in the smallest calibration model error. An increase in $|k|$ from three to four did not result in a large difference in calibration model error (20.4×10^{-3} m for a $|k|$ of three and 20.5×10^{-3} m for a $|k|$ of four). So, the optimal calibration model was nonlinear with a $|k|$ of three and a fixed principal point and aspect ratio. Calibration model error for this calibration model was 20.5×10^{-3} m.

3.3.3 V3

This was the non-perpendicular and non-fisheye viewpoint.

2D-DLT gave the highest calibration model error (46.1×10^{-3} m). Nonlinear calibration models that had a $|k|$ of zero and estimated tangential distortion had the lowest calibration model error (38.9×10^{-3} m to 40.2×10^{-3} m). There were only small differences in calibration model error between the four models that met these criteria. So, the optimal calibration model was nonlinear with a $|k|$ of zero, calculated tangential distortion and fixed principal point and aspect ratio. Calibration model error for this calibration model was 39.3×10^{-3} m.

3.3.4 V4

This was the perpendicular and non-fisheye viewpoint. It was based on a camera location used by British Swimming at the 2012 Olympic Games.

2D-DLT gave a calibration model error that was higher than 24 of the 32 nonlinear calibration models (10.3×10^{-3} m for 2D-DLT and 14.9×10^{-3} m for the highest nonlinear). Nonlinear models with a $|k|$ of two or three had the lowest calibration model error (4.3×10^{-3} m to 12.4×10^{-3} m). Of these, calibration models that calculated tangential distortion and fixed aspect ratio had, in general, the lowest calibration model error (4.3 x 10^{-3} m to 4.6 x 10^{-3} m). There were only small differences in calibration model error between the four models that met these criteria. So, the optimal calibration model was nonlinear with a |*k*| of two, calculated tangential distortion and fixed principal point and aspect ratio. Calibration model error for this calibration model was 4.6 x 10^{-3} m.

3.3.5 V5

This was the multi-camera viewpoint.

2D-DLT gave the highest calibration model error (14.6 x 10^{-3} m) followed by the nonlinear calibration models with a |*k*| of zero and one (9.9 x 10^{-3} m to 14.3 x 10^{-3} m). Once these models were discounted then fixing the principal point and aspect ratio and not calculating tangential distortion gave two of the lowest three calibration model errors. A |*k*| of three gave the smallest calibration model error (7.2 x 10^{-3} m). Increasing |*k*| from two to three did not result in a large increase in calibration model error (7.7 x 10^{-3} m for a |*k*| of two). So, the optimal calibration model was nonlinear with a |*k*| of two, no tangential distortion and a fixed principal point and aspect ratio. Calibration model error for this calibration model was 7.7 x 10^{-3} m.

3.3.6 Summary

The optimal calibration models for each viewpoint are summarised in Table 3.3. All of these used the nonlinear calibration procedure.

Table 3.3. Summary of the optimal calibration models.

	Viewpoint				
	V1	V2	V3	V4	V5
|*k*|	1	3	0	2	2
Calculate tangential?	N/A	N/A	Yes	Yes	No
Calculate principal point?	Yes	No	No	No	No
Calculate aspect ratio?	No	No	No	No	No
Reprojection error *n*	1296	1332	1296	1188	1332
Reprojection error (pixels)	0.60	0.80	0.51	0.46	0.52
RDE *n*	56	56	56	56	20
RDE mean (10^{-3} m)	10.0	5.7	-6.3	0.0	-0.2
RDE std. dev. (10^{-3} m)	12.6	19.8	39.1	4.7	7.9
Calibration model error (10^{-3} m)	16.0	20.4	39.3	4.6	7.7
Distortion *n*	144	144	144	144	144
Distortion max. (pixels)	263.3	267.4	2.8	6.5	3.4

Figure 3.7 shows how absolute RDE varied across the calibrated plane for each viewpoint. In general, a regular pattern was not found but V3 had a weak trend toward higher RDE as distance from the camera increased. The two adjacent large RDE for this viewpoint at 25 m in the Y axis were both underestimations of the ground truth distance: the test distances were reconstructed as 4.88 m and 4.91 m. So, it was not a single error in the reconstruction of their shared point at (35, 25, 0) that caused these two large RDE.

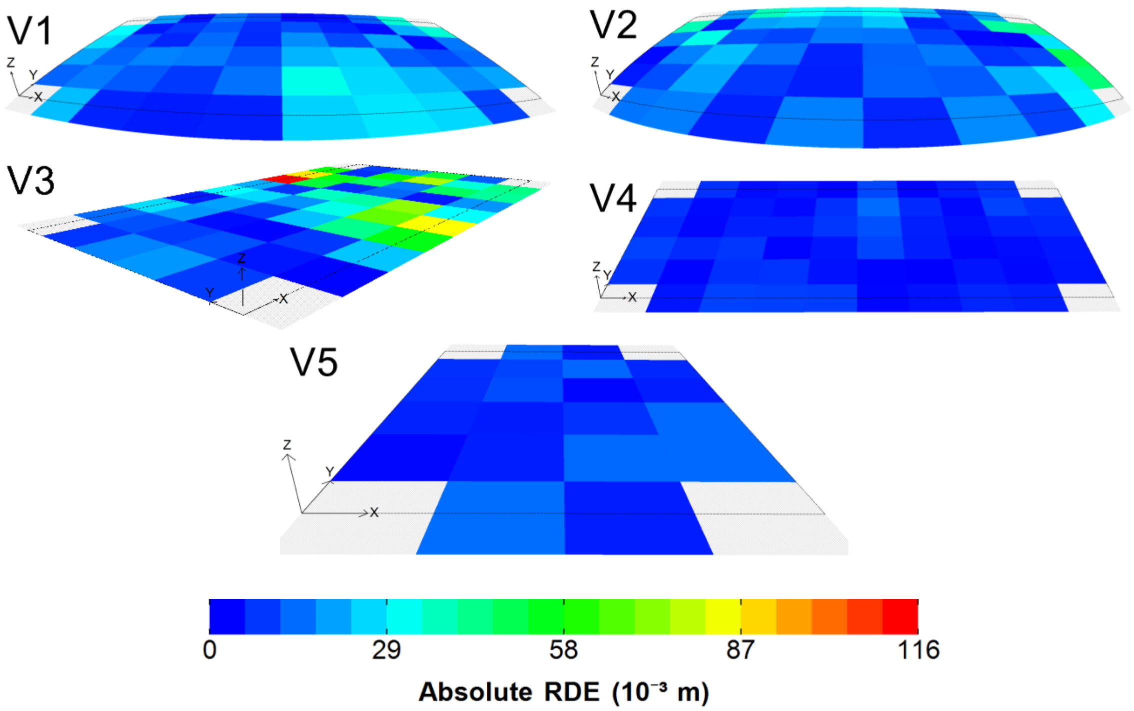


Figure 3.7. Pattern of absolute RDE. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed.

Figure 3.8 shows the pattern of distortion for each viewpoint. In general, the lowest distortion was in the centre of the image and the highest toward the edges.

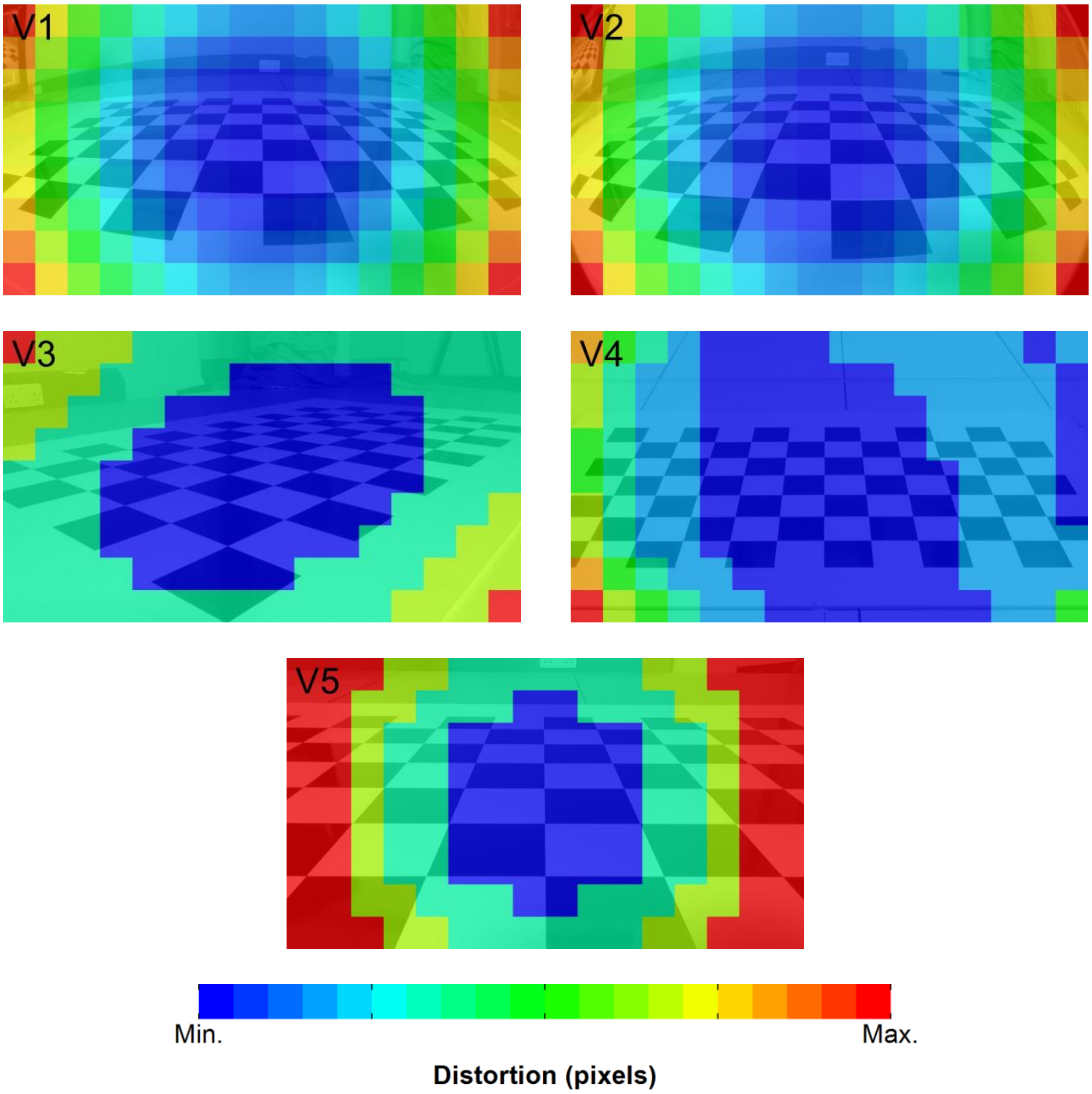


Figure 3.8. Pattern of distortion. Each heat map had its own scale with blue representing the viewpoint's minimum and red the maximum.

3.4 Discussion

3.4.1 Comparison of single- and multi-camera viewpoints

V1 to V4 were the single-camera viewpoints and V5 was the multi-camera viewpoint.

Calibration model error for V1 to V4 was lower than that found for the benchmark multi-camera viewpoint. Veiga et al. (2010) reported a value of 46 x 10^{-3} m. V1 to V4 were in the range 5 x 10^{-3} m to 39 x 10^{-3} m. This was an unexpected result: intuition suggested that increasing the size of the calibrated plane in the X axis should have led to an increase in calibration model error.

A key factor in this was the difference in methods. Veiga et al. (2010) chose a real swimming pool whereas this study used a checkerboard scale swimming pool. The scale swimming pool had highly accurate and precise control point and test distance coordinates. This isolated the calibration model error, i.e. RDE was due to an incorrect relationship between distances in the pixel and world coordinate system rather than errors in the measurement of the control points or test distances. This may not have been the case for a real swimming pool; for example, the test distances, being located on the floating lane ropes, may have had changeable world coordinates. Furthermore, the checkerboard scale swimming pool used a semi-automated and sub-pixel accurate method of calculating pixel coordinates; such a method was probably not possible with a real swimming pool. Therefore, it is likely that the calibration model error found by Veiga et al. (2010) also contained error in the measurement of the control points and test distances. How such uncertainties affect reconstruction is the subject of Chapters 5, 6, 7, 8, 10 and 11.

An additional factor was image size. V1 to V4 used FHD footage, i.e. images that were 1920 pixels wide and 1080 pixels high. Veiga et al. (2010) did not explicitly state the size of their images. Their camera, a JVC GY-DV500E (JVC 2014), recorded to Digital Video (DV) format, which typically gives Standard-Definition (SD) images that are 720 pixels wide by 576 pixels high. So, although V1 to V4 had a larger calibrated plane in the X axis than Veiga et al. (2010), the image captured of it was also larger in width and height. As a result, the disparity in resolution (i.e. the number of pixels that represented a metre) between the images for V1 to V4 and Veiga et al. (2010) was lower than if the same resolution were used in both cases. The differences in size between FHD and SD images are highlighted in Figure 3.9.

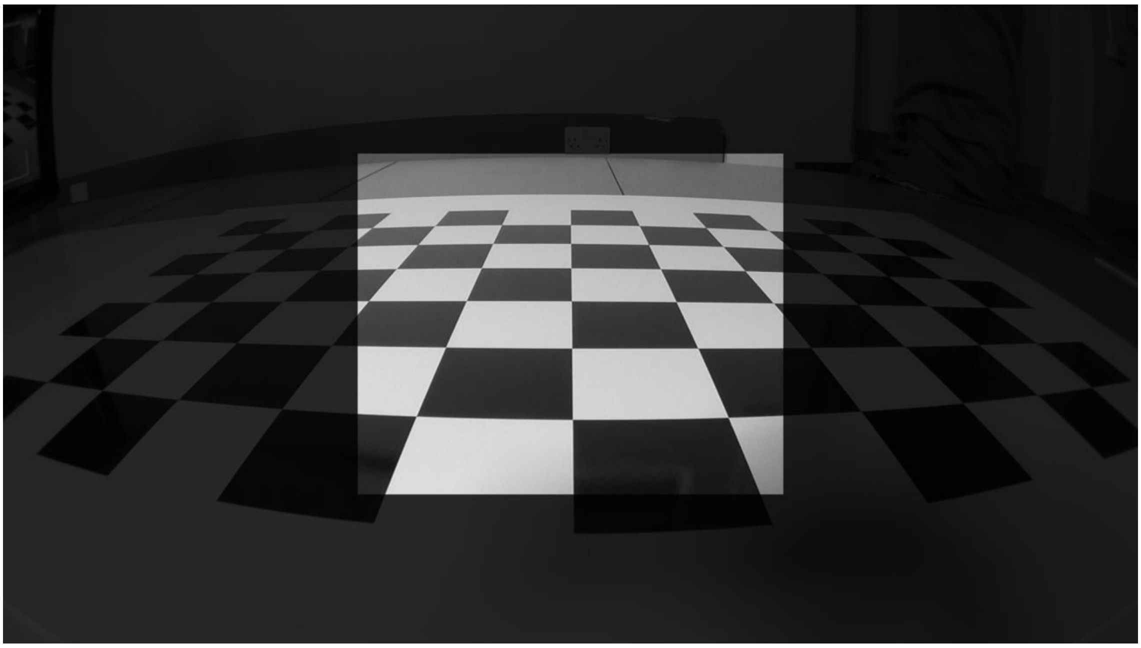


Figure 3.9. An SD sized viewport superimposed on a FHD image of the calibrated plane taken from V2.

The region of the calibrated plane inside the viewport in Figure 3.9 was similar in size to the calibrated plane used by Veiga et al. (2010), i.e. between 10 m and 20 m in the swimming direction for all swimming lanes. This gave a visual confirmation that the resolution for V1 to V4 was similar to what Veiga et al. (2010) had. Resolution is examined in more detail in Chapter 4.

V5 used the same experimental set-up and image size as the single-camera viewpoints. If calibration model error were positively related to the size of the calibrated plane in the X axis, then V5 should have had the lowest calibration model error. This was not the case: it was 4.6 x 10^{-3} m for V4 and 7.7 x 10^{-3} m for V5. It was not clear what led to this result. A possibility is that V4 and V5 are examples of an underlying limit on the accuracy with which distances could be reconstructed by the nonlinear calibration procedure and the difference between them was due to factors such as noise in the world and image coordinates of the control points and test distances. The results showed that increasing the size of a calibrated plane in the X axis does not inevitably lead to an increase in calibration model error.

3.4.2 Comparison of 2D-DLT and nonlinear calibration models

For all viewpoints, the 2D-DLT calibration model was bettered by a nonlinear one. A key difference between them was the additional information about the relationship between pixel and world coordinate systems that the nonlinear models got from the images of the calibration object. The 2D-DLT models had only the world and pixel coordinates of the control points to establish this relationship.

V1 and V2 were fisheye viewpoints.

The difference between the 2D-DLT and nonlinear calibration models was most apparent for V1 and V2. For these two viewpoints, the calibration model error was 1018.3 x 10^{-3} m to 1353.5 x 10^{-3} m for the 2D-DLT models and 16.0 x 10^{-3} m to 20.4 x 10^{-3} m for the optimal nonlinear models. To understand why this was the case the 2D-DLT calibration model for V1 was examined in more detail. The distance between C1 and C2 (i.e. at 0 m in the Y axis) was 1767 pixels so a linear mapping between pixels and metres would make each 5 m test distance about 177 pixels. Although the actual lengths ranged from 126 pixels to 264 pixels two were close to 177 pixels. This is shown in Figure 3.10.

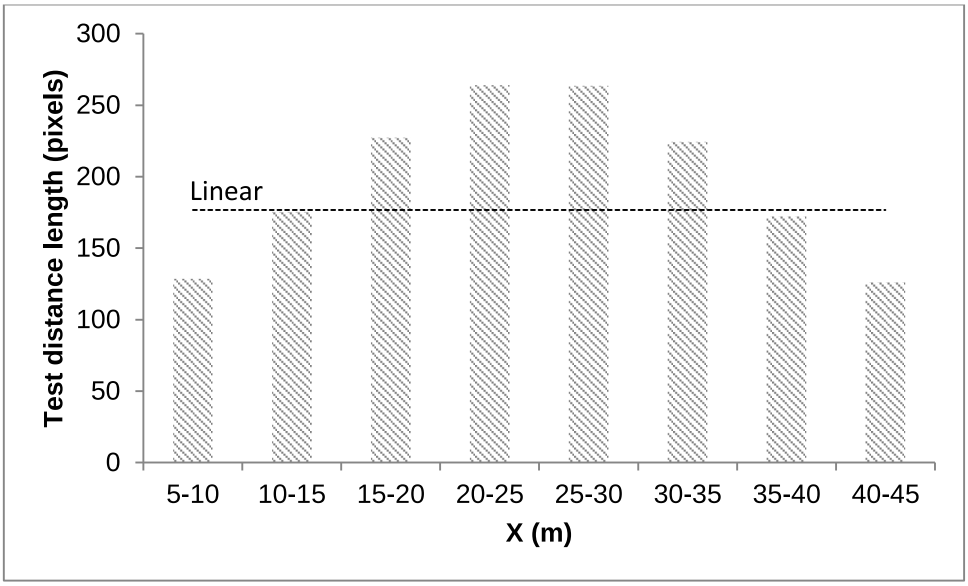


Figure 3.10. Test distance length in pixels between control points C1 and C2 for V1. A linear mapping between pixels and metres made each 5 m test distance about 177 pixels.

A test distance's RDE was positively correlated with the disparity, in pixels, between the length predicted by a linear mapping and its actual length. This is shown in Figure 3.11. The two test distances whose length was similar to the linear mapping value had the smallest RDE and the RDE for the others reflected how far they were from the linear mapping length.

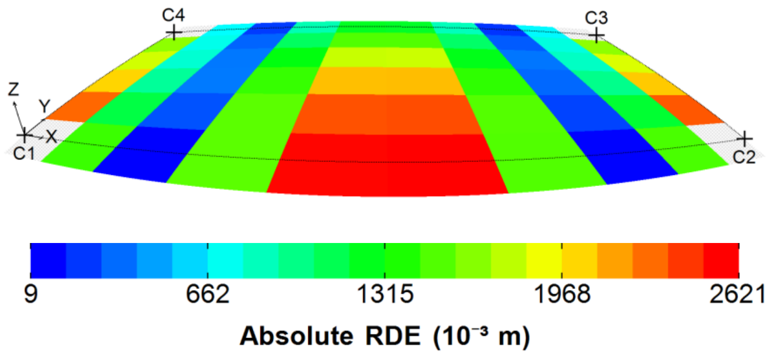


Figure 3.11. Pattern of absolute RDE for the 2D-DLT calibration model for V1. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed.

3.4.3 Comparison of single-camera viewpoints

V1, V2 and V4 were the perpendicular viewpoints and V3 the non-perpendicular viewpoint.

V3 had a noticeably higher calibration model error than did V1, V2 and V4. Figure 3.7 showed that the highest RDEs for V3 tended to be for the test distances that were furthest from the camera. As will be shown in Chapter 4, these test distances were shorter in pixels than those that were closer to the camera and those in V1, V2 and V4. This suggested that RDE was negatively correlated with a test distance's length in the pixel coordinate system. In conclusion, that V3 was a non-perpendicular viewpoint led to a predominance of short test distances and this led to high calibration model error.

3.5 Conclusion

This chapter calculated calibration model error and found:

- Calibration model error for four single-camera viewpoints was lower than that reported in the literature for a multi-camera viewpoint.

- Increasing the size of the calibrated plane in the swimming direction did not always lead to higher calibration model error.
- A correctly chosen nonlinear calibration model produced lower calibration model error than did 2D-DLT.
- If the swimming competition venue allowed it, the best viewpoint was perpendicular and non-fisheye. This viewpoint had the lowest calibration model error.
- A perpendicular and fisheye viewpoint should be chosen over a non-perpendicular and non-fisheye one.

The next chapter highlights a potential disadvantage of using a single-camera viewpoint instead of a multi-camera one: resolution, i.e. the number of pixels per metre.

4 Resolution

4.1 Introduction

This chapter calculated the relationship between the size of objects in the pixel and world coordinate systems. This relationship was called resolution and was defined as the number of pixels per metre. Two objects with different resolutions are shown in Figure 4.1.

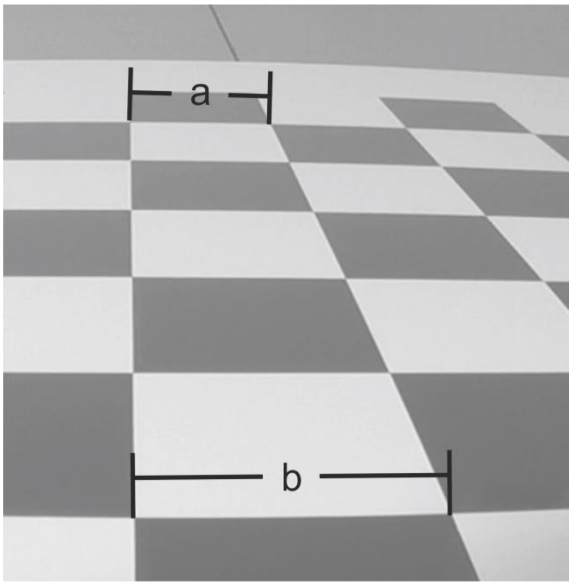


Figure 4.1. Distances a and b were the same length in metres but different lengths in pixels. So, they had different resolutions.

Resolution was calculated for the single-camera viewpoints described in Chapter 3. These results were compared to two multi-camera viewpoints: one that represented the viewpoints described in the competition performance analysis literature (e.g. Arellano et al. 1994, Hellard et al. 2008) and a viewpoint that used the same experimental set-up as the single camera ones.

4.2 Method

The number of pixels per metre (pixels m^{-1}) was calculated for the viewpoints introduced in Chapter 3. The following steps were done for each corner on the checkerboard scale swimming pools shown in Figure 3.5:

1. Line endpoints ± 0.5 m in the X and Y axis of the ground truth world coordinates of the corner were projected using Equations 2.7 to 2.17 and the optimal nonlinear calibration models listed in Table 3.3. An example of this projection is shown in Figure 4.2.

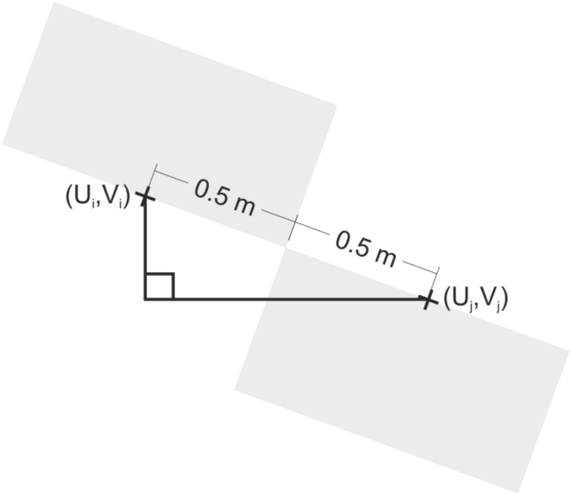


Figure 4.2. Points were projected ± 0.5 m of a grid square corner. The length of the line in pixels was the hypotenuse of a right-angled triangle, whose opposite and adjacent were aligned with the pixel coordinate system's U and V axes.

2. Calculated the length of these lines in pixels using trigonometry to give pixels m^{-1} in the X and Y axes.
3. Calculated the resultant pixels m^{-1}, i.e. the square root of the sum of the squared X and Y pixels m^{-1}.

V5 was a FHD and multi-camera viewpoint.

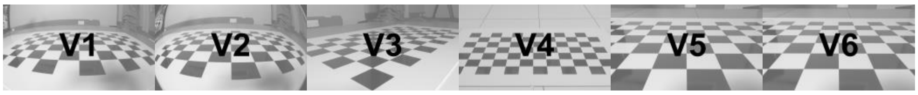
A Standard Definition (SD) viewpoint, called V6, was created. It was based on V5. Pixels m^{-1} for V6 was calculated using the steps described above. At step 1 the ground truth world coordinates, intrinsic parameters and distortion coefficients for V5 were used. Then, at step 2, pixels m^{-1} in the X axis was multiplied by 720/1920 and pixels m^{-1} in the Y axis was multiplied by 576/1080. This converted the pixels m^{-1} values from FHD to SD. In this way, the resolutions for V6 were representative of that found in the competition performance analysis literature (e.g. Arellano et al. 1994, Hellard et al. 2008).

4.3 Results

The mean ± standard deviation of the resolution in the X axis and resultant direction is in Table 4.1. V5 had the highest resolutions. V6 had resolutions that were similar to, but slightly lower than, V1 to V4.

Table 4.1. Mean ± standard deviation of the resolution in the X axis and resultant direction.

Viewpoint	n	X axis (pixels m^{-1})	Resultant (pixels m^{-1})
V1	66	26.4 ± 8.9	32.0 ± 10.0
V2	66	27.5 ± 10.9	33.3 ± 11.5
V3	66	23.2 ± 11.8	37.3 ± 11.4
V4	66	30.7 ± 2.6	34.5 ± 3.6
V5	30	59.0 ± 16.9	65.5 ± 22.3
V6	30	22.1 ± 6.3	26.8 ± 10.1



In general, resolution in the X axis reduced with increasing distance in X and Y from the camera location. This is illustrated in Figure 4.3.

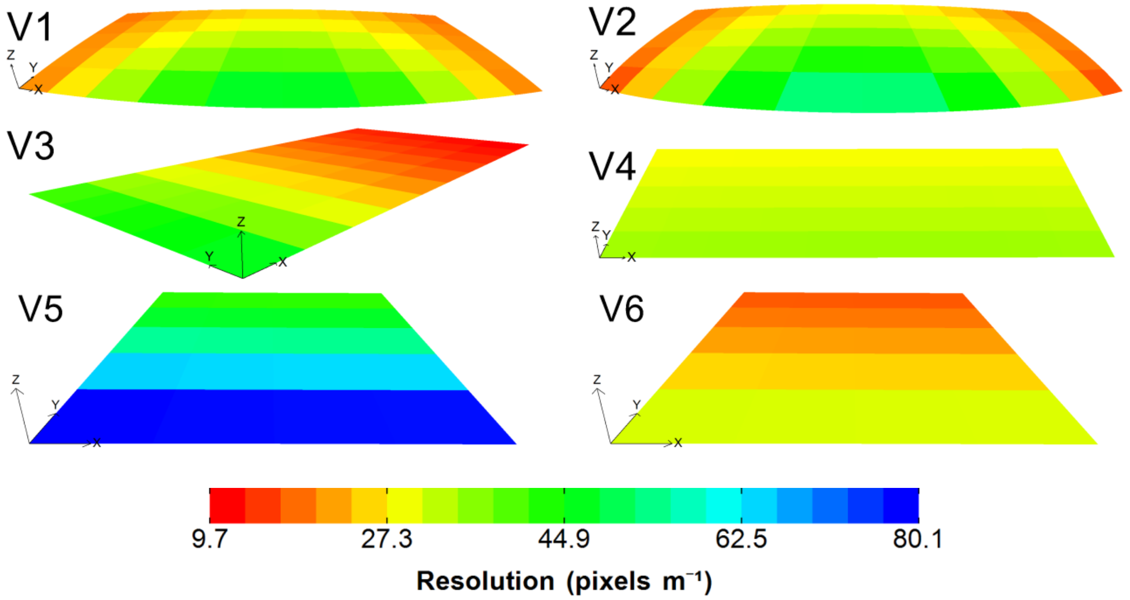


Figure 4.3. Pattern of resolution in the X axis. The colour of each heat map square represented the mean of the pixels m^{-1} calculated for its corners. As a result, $n = 4$ for each of the squares in each heat map.

For all viewpoints, resolution in the resultant direction had a similar pattern to that found for the X axis but V1 and V2 had larger regions of low resolution whereas V3 had a smaller one. This is illustrated in Figure 4.4.

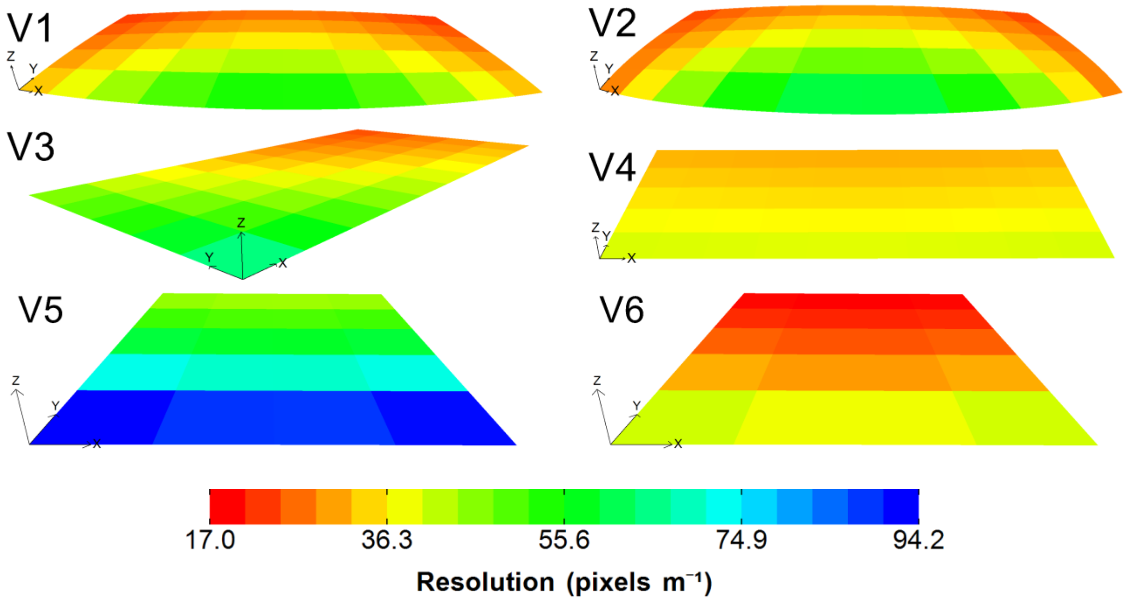
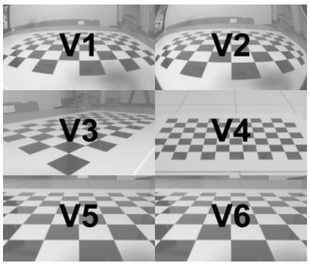


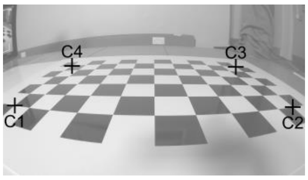
Figure 4.4. Pattern of resolution in the resultant direction. The colour of each heat map square represented the mean of the pixels m^{-1} calculated for its corners. As a result, n = 4 for each of the squares in each heat map.

Resolution in the resultant direction at the four control points is given in Table 4.2. In general, it reduced with increasing distance in X and Y from the camera location. This was not the case for V2: it was similar for all control points.

Table 4.2. Resolution in the resultant direction for the four control points, Ci (i = 1 to 4).

Viewpoint	Resultant (pixels m^{-1})			
	C1	C2	C3	C4
V1	30.9	30.5	19.9	20.0
V2	22.8	22.8	20.6	20.6
V3	74.5	30.8	22.1	47.3
V4	40.4	40.5	30.0	29.9
V5	110.4	109.3	41.7	42.0
V6	48.2	47.9	16.4	16.5





4.4 Discussion

V1 to V5 were FHD viewpoints and V6 was the SD viewpoint. V1 to V4 were single-camera viewpoints and V5 and V6 were multi-camera viewpoints.

As expected, for a constant image size, increasing the size of the calibrated plane in the swimming direction led to a decrease in pixels m^{-1}. For example, Figure 4.3 and Figure 4.4 showed that pixels m^{-1} in the X axis and resultant direction across the entire calibrated plane for V4 were less than the lowest pixels m^{-1} for V5.

Clearly, when the calibrated plane and image size were both increased by a similar scale, as was the case when V1 to V4 are compared to V6, then the pixels m^{-1} remained largely unchanged. The typical viewpoint reported in the literature is perpendicular, non-fisheye, multi-camera and SD. In this study, V6 was used to approximate this. It had a calibrated plane that was 20 m in the X axis, i.e. 40% of the size of V1 to V4. It also had an image width that was 720 pixels, which was 37.5% of that for V1 to V4. This led to similar resolutions for V6 and V1 to V4, which showed that increasing the size of the calibrated plane in the X axis was compensated for by increasing the image size.

A key cause of the pattern of resolution was perspective projection. An outcome of this type of projection is that an object that is close to the camera appears larger in the pixel coordinate system than it would if it were further from the camera. This led to a negative correlation between pixels m^{-1} and distance from the camera. This effect is clearest for V5 and V6 in Figure 4.3 and Figure 4.4 and was caused by Equation 2.10, i.e. the transformation from world to camera coordinate systems, which is a perspective projection.

V1 and V2 were the fisheye viewpoints.

Another factor combined with perspective projection to cause the pattern of resolution for V1 and V2: distortion. The barrel distortion caused by the fisheye lens converter decreased magnification as the Euclidean distance from the principal point increased. This meant that toward the edges of the calibrated plane more of the world was fitted into the same sized area of the image. The result of this was low pixels m^{-1} in the X axis and resultant direction in the regions of highest distortion.

The combined effect that perspective projection and distortion had on pixels m^{-1} in the X axis and resultant direction was most apparent in the regions around control points C1 and C2 for V2. When considering perspective projection, the pixels m^{-1} near to C1 and C2 would be equal to or greater than anywhere on the rest of the calibrated plane. The distortion near C1 and C2 was so extreme that the pixels m^{-1} near these control points was actually low compared to the rest of the calibrated plane.

4.5 Conclusion

This chapter examined the effect that single and multi-camera viewpoints had on the number of pixels that represented a metre on the calibrated plane, and found:

- For a constant image size, increasing the size of the calibrated plane in the swimming direction led to a decrease in pixels m^{-1}. As a result, pixels m^{-1} for a single-camera viewpoint would be lower than it was for a multi-camera one.
- Increasing the size of the image of a calibrated plane from SD to FHD compensated for an increase in the size of the calibrated plane in the X axis from 20 m to 50 m. This made the pixels m^{-1} for a single-camera viewpoint comparable with that found in the competition performance analysis literature.
- Perspective projection and distortion are key factors in pixels m^{-1}.

In the next chapter the effect of uncertainty in the control point coordinates on RDE is examined. It will be shown that pixels m^{-1}, especially in the regions around the control points, was a key factor in this.

5 Control point error

5.1 Introduction

This chapter calculated control point error. This was the error in reconstructing a distance that was caused by uncertainty in the measurement of the control points used to calculate a calibration model. Figure 5.1 shows its place in the hierarchy of errors; all the other errors in this hierarchy were fixed at zero in this chapter.

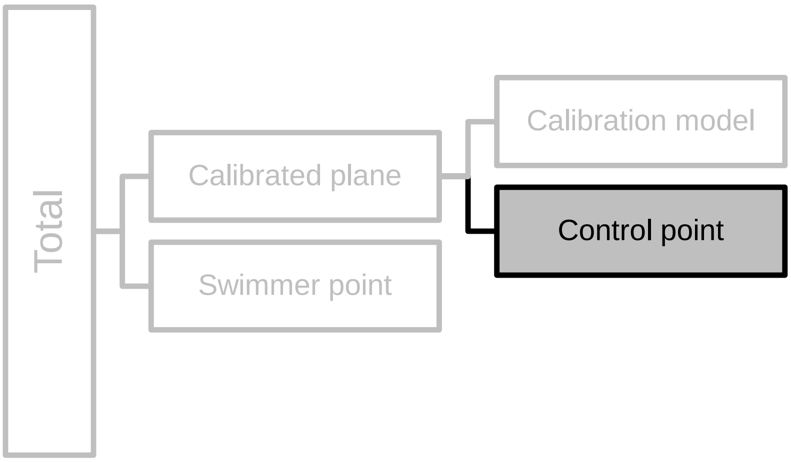


Figure 5.1. Chapter 5 calculated control point error. Calibration model and swimmer point errors were fixed at zero.

Control point error was calculated using a computer simulation of the four single-camera viewpoints and scale swimming pool introduced in Chapter 3.

5.2 Method

The following steps were performed. Each of these steps is described in more detail in the following subsections.

1. A way of simulating the uncertainty in the measurement of a control point's world and pixel coordinates was established. For this,

distributions that described the likely uncertainty were created. See the **World coordinate system uncertainty** and **Pixel coordinate system uncertainty** sections.

2. Control point error was calculated using a computer simulation of the four single-camera viewpoints and scale swimming pool introduced in Chapter 3. See the **Simulation** section.

5.2.1 World coordinate system uncertainty

Uncertainty in the measurement of a control point's X coordinate was modelled as a Gaussian half-normal distribution. A half-normal one was chosen as a result of FINA rule FR 2.2.1 (FINA 2013a), which stated that a competition pool must measure at least 50 m in this axis. As a result, the half normal distribution ensured that uncertainty in an X coordinate only led to a distance of 50 m or more. Uncertainty in the measurement of a control point's Y coordinate was modelled as a Gaussian normal distribution; the FINA rules did not provide tolerances for measurements in this axis.

The standard deviation of these distributions was based on the precision of a low-cost laser distance measurement device: a Leica Geosystems DISTRO D2. The device's standard deviation was 0.750 x 10^{-3} m for measurements up to 10 m with a maximum increase of 0.075 x 10^{-3} m per metre for distances above this (Leica Geosystems 2014). So, the X coordinate distribution was given a standard deviation of 3.750 x 10^{-3} m and the Y coordinate distribution a standard deviation of 1.875 x 10^{-3} m. Both distributions had a mean of 0 x 10^{-3} m.

These distributions were used to simulate the measurement of a control point. Four measurements were simulated: two in the X axis and two in the Y axis.

This reflected how the measurements would be taken in reality, i.e. the measurement of a distance between pairs of adjacent control points in a certain axis. A control point's world coordinates were simulated using the steps described below and shown in Figure 5.2.

The process for simulating the measurement in the X axis between control points i and j was:

1. Initialised X_i to 0 m and X_j to 50 m. This gave the ground truth coordinates.
2. Drew a random sample ε from a half-normal Gaussian distribution with a mean of 0 x 10^{-3} m and a standard deviation of 3.750 x 10^{-3} m.
3. Drew a random sample f from a uniform distribution with a range of zero to one.
4. Translated X_i and X_j along the X axis by $(-f)\varepsilon$ and $(1-f)\varepsilon$ respectively.

The process for simulating the measurement in the Y axis between two control points i and j was:

1. Initialised Y_i to 0 m and Y_j to 25 m. This gave the ground truth coordinates.
2. Drew a random sample ε from a normal Gaussian distribution with a mean of 0 x 10^{-3} m and a standard deviation of 1.875 x 10^{-3} m.
3. Drew a random sample f from a uniform distribution with a range of zero to one.
4. Translated Y_i and Y_j along the Y axis by $(-f)\varepsilon$ and $(1-f)\varepsilon$ respectively.

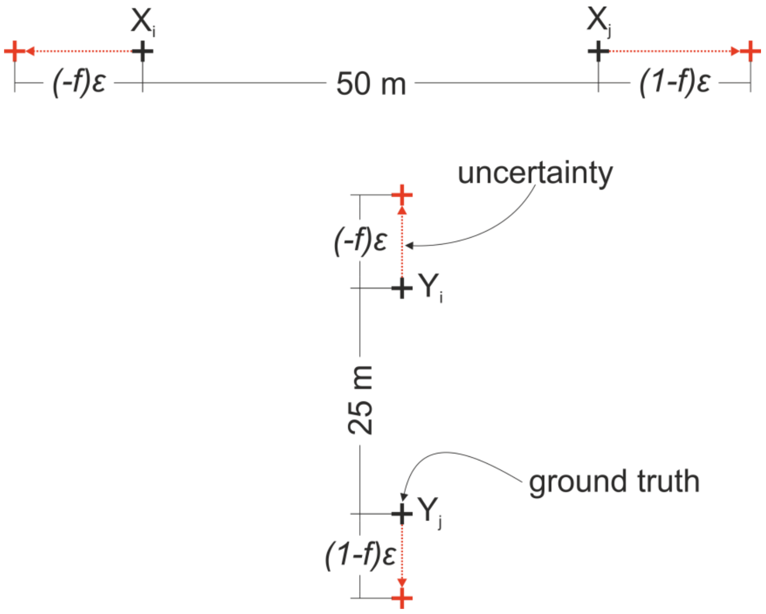


Figure 5.2. Method of simulating the measurement of a control point in the X and Y axes.

5.2.2 Pixel coordinate system uncertainty

An assessment of intra-analyst precision in control point digitisation was carried out. Four control points in two images of a 50 m by 25 m swimming pool were repeatedly digitised by one analyst. The control points were the corners of the swimming pool at the water surface. The images were captured from two single-camera viewpoints (one fisheye and one non-fisheye). Each control point in each image was manually digitised twenty times with at least twenty-four hours between repeats. The mean U and V coordinate of each control point was taken as the best estimate of its ground truth pixel coordinates. The digitised points were randomly spread about the mean; this is shown for two of the control points, one from each image, in Figure 5.3.

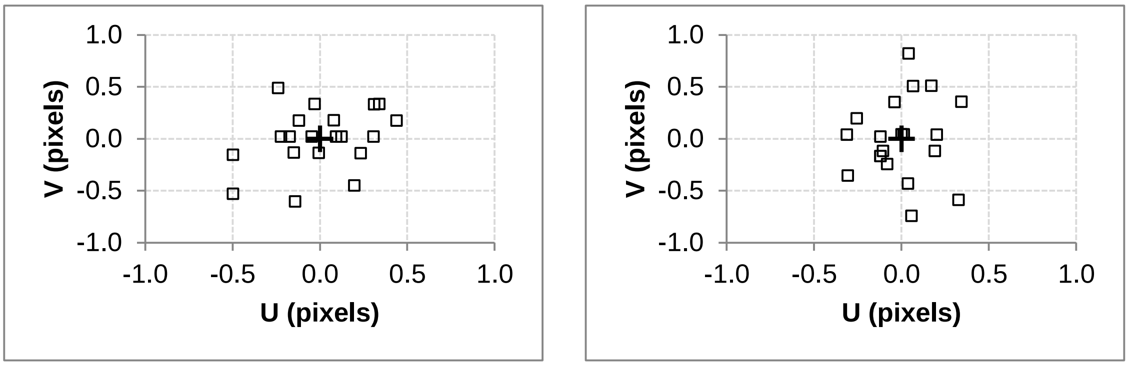


Figure 5.3. Visualisation of the repeated digitisation of two control points. The cross is the mean of the digitisations. A square shows the difference between an individual digitisation and the mean.

The Euclidean distance between each digitised coordinate and the mean for the corresponding control point was calculated. The standard deviation of these distances was 0.19 pixels to two decimal places. This was used to create a distribution that described the uncertainty in the measurement of a control point's U and V coordinates. A Gaussian normal distribution was used. It had a mean of zero pixels and a standard deviation of 0.19 pixels.

Four measurements in the pixel coordinate system were simulated: one for each control point. The process for simulating a measurement of control point i was:

1. Initialised U_i and V_i to their ground truth value.
2. Drew a random sample ε from a normal Gaussian distribution with a mean of 0 pixels and a standard deviation of 0.19 pixels.
3. Drew a random sample α from a uniform distribution with a range of 0° to 359°.
4. Translated U_i along the U axis by $\varepsilon \cos \alpha$ and V_i along the V axis by $\varepsilon \sin \alpha$.

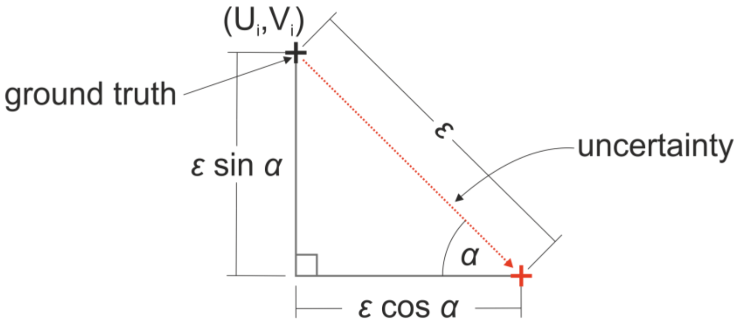


Figure 5.4. Simulating uncertainty in the measurement of a control point in the U and V axes.

5.2.3 Simulation

A computer simulation of the four single-camera viewpoints (i.e. V1 to V4) and scale swimming pool introduced in Chapter 3 was developed. The coordinates of the control points and the test distance endpoints for each viewpoint were based on those used in Chapter 3. Test distances in this chapter had projected pixel coordinates; this removed calibration model error, i.e. initial RDE was zero.

Two scenarios were simulated for each viewpoint. In the first, uncertainty was added to the world coordinates of the four control points, but not the pixel coordinates. The second did the opposite. These are referred to as the world and pixel scenarios. Ten thousand independent trials were done for each scenario: each one measured the control points (using the methods illustrated in Figure 5.2 and Figure 5.4), calculated a nonlinear calibration model, reconstructed the test distances, and calculated their RDE. The simulation used the intrinsic parameters and distortion coefficients of the optimal nonlinear calibration models listed in Table 3.3.

5.3 Results

The mean ± standard deviation of the control point error is shown in Table 5.1. The standard deviation of the control point error was much larger in the pixel scenario than it was in the world scenario. Control point error had zero bias for the pixel scenario while the world scenario had a small bias.

Table 5.1. Mean ± standard deviation of the control point error.

Scenario	n	V1	V2	V3	V4
World (10^{-3} m)	560000	0.2 ± 0.3	0.3 ± 0.3	0.3 ± 0.3	0.3 ± 0.2
Pixel (10^{-3} m)	560000	0.0 ± 4.6	0.0 ± 4.5	0.0 ± 2.8	0.0 ± 0.7

5.4 Discussion

Four control points were used when simulating the calculation of a nonlinear calibration model. This was the minimum required to do this, which meant that the solution for the extrinsic parameters was not over-determined and error minimisation was not possible. As a result, uncertainty in the four control points propagated to error in the reconstructed distances.

The mechanism with which it propagated was not straightforward. Consider the trial that led to eight of the ten largest control point errors for its viewpoint. It did not have the largest control point uncertainties: over 10% of the trials had larger ones. In addition, the largest uncertainties did not act mainly in the swimming direction. On the contrary, as shown in Figure 5.5, they acted perpendicular to it. This was counter to intuition: the test distances were in the X axis and it was expected that the largest control point errors would have resulted from control point uncertainties that also acted in this axis.

This highlighted why a computer simulation method was chosen. For a nonlinear calibration model, it was difficult to predict how uncertainty in control point measurements would propagate to control point error. So, ten thousand trials were simulated. This quantified the systematic and random error in reconstructed distances that would be caused by measurement of the swimming pool and digitisation of the control points.

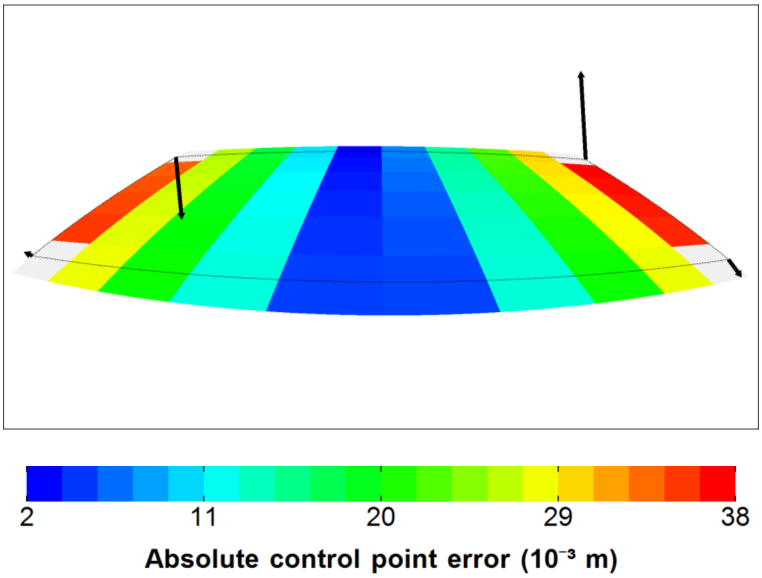


Figure 5.5. Trial that had eight of the ten largest control point errors for its viewpoint. Black arrows show the uncertainty in the control point's pixel coordinates; these were scaled 400:1 for illustration purposes. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed.

In general, control point error was small. The mean of the control point error was less than 0.3 x 10^{-3} m and the standard deviation was less than 5 x 10^{-3} m, i.e. under 0.1%. So, errors in reconstructed distances caused by uncertainty in the measurement of control points were likely to be small.

The pixel scenario had higher variability in the control point error than did the world scenario. For three out of the four viewpoints, this disparity was an order of magnitude or more. So, the current method used to measure the pool, i.e. a low-cost laser distance measurement device, was deemed fit for purpose. In contrast, control point digitisation could be improved. Such improvements may lead to reductions in control point error and hence improved precision when reconstructing distances.

5.5 Conclusion

This chapter calculated control point error and found, for four single-camera viewpoints, which used nonlinear calibration models to reconstruct distances:

- Confirmation that uncertainty in a control point's coordinates would propagate to error in reconstructed distances. This error was likely to be small. So, when using a nonlinear calibration model to reconstruct distances, uncertainties in the measurement of the swimming pool and digitisation of the control points were likely to have a small impact on the accuracy and precision of the reconstructed distances.
- Measuring a 50 m by 25 m swimming pool using a low-cost laser distance measurement device was deemed an appropriate method.
- An improved method of determining a control point's pixel coordinates could be beneficial.

This last point provided the motivation for the next chapter in which an improved method of determining a control point's pixel coordinates was developed. Then, in Chapter 7, the effect that this new method had on control point error was calculated.

6 Reducing uncertainty in control point pixel coordinates

6.1 Introduction

The previous chapter showed that uncertainty in a control point’s world and pixel coordinates propagated to error in reconstructed distances. In general, these errors were small. However, it was identified that an improvement to the current method of determining a control point’s pixel coordinates had the potential to make them even smaller.

Due to the size of the error in reconstructed distances caused by control point digitisation, the new method needed to be quick and easy as a time-consuming or complex method may not have been worth the marginal improvements it may achieve. The new method developed in this chapter achieved this by using the physical features of a swimming pool and a simple error minimisation technique.

In an undistorted image of a swimming pool, the water surface is bound by four straight lines. If the control points were at the corners of the swimming pool, then they would be at the intersections of these lines. Furthermore, if more than two points on each line were given, simple linear regression, which minimises error in a least squares sense, could be used to calculate each line’s equation and hence the intersections. These steps were the basis of the new method, which was called the line-line method and is illustrated in Figure 6.1. It was postulated that the line-line method would improve the precision of a control point’s pixel coordinates compared to the current method. This was tested in a computer simulation.

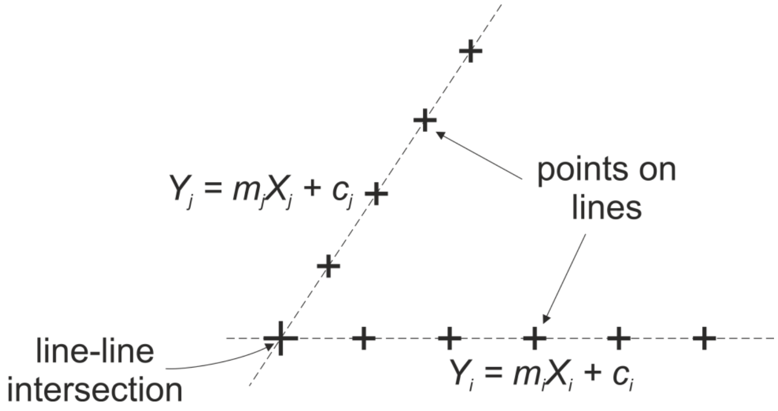


Figure 6.1. Line-line method for calculating the pixel coordinates of a control point, which was at the intersection of lines that bounded a swimming pool.

6.2 Method

The following steps were performed. Each of these steps is described in more detail in the following subsections.

1. A way of simulating the uncertainty in the digitisation of a point on a swimming pool's boundary was established. For this, distributions that described the likely uncertainty were created. See the **Pixel coordinate system uncertainty** section.
2. Line-line error was calculated using a computer simulation of the four single-camera viewpoints and scale swimming pool introduced in Chapter 3. See the **Simulation** section.

6.2.1 Pixel coordinate system uncertainty

Uncertainty in the digitisation of points on a swimming pool's boundary was modelled as a Gaussian normal distribution with a mean of 0 pixels and a standard deviation of 0.19 pixels. The same distribution was used to add uncertainty to the digitisation of a control point in Chapter 5.

6.2.2 Simulation

A computer simulation of the four single-camera viewpoints (i.e. V1 to V4) and scale swimming pool introduced in Chapter 3 was developed. The coordinates of the points on the boundary of the scale swimming pool for each viewpoint were based on those used in Chapter 3. These are shown in Figure 6.2. The points on the boundary had projected pixel coordinates; this removed calibration model error.

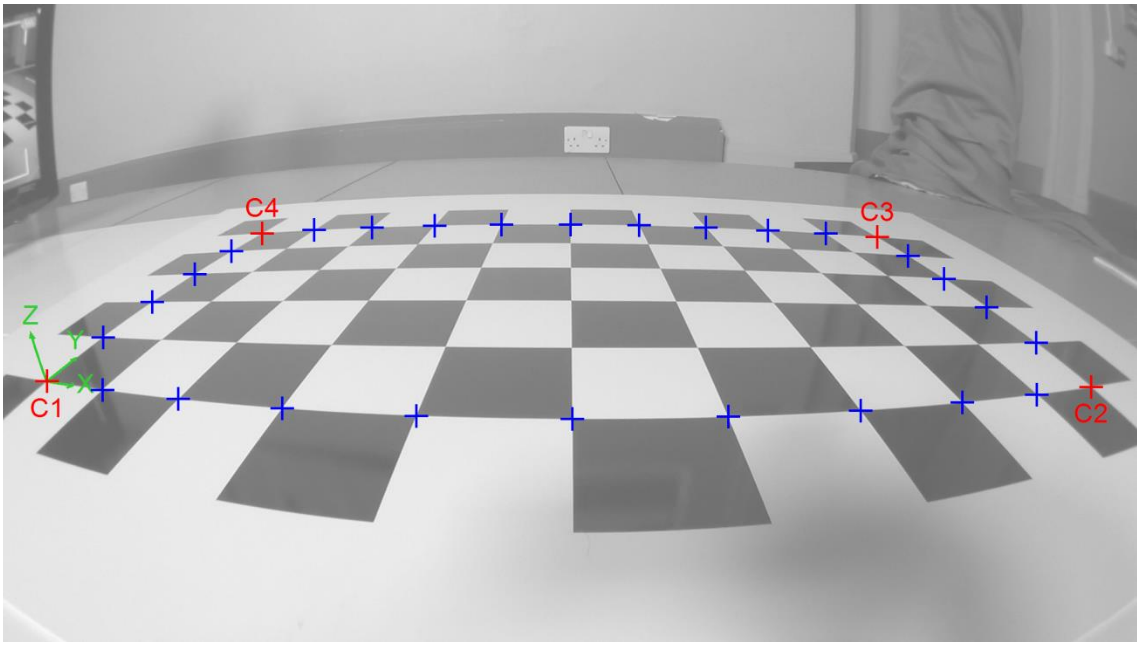


Figure 6.2. Points on the scale swimming pool's boundary. The intersections of these lines were used to estimate the pixel coordinates of control points, Ci (i = 1 to 4).

Ten thousand independent trials were done for each viewpoint. Each trial did the following steps for each of the four control points:

1. Added a randomly selected uncertainty to each point on the two intersecting lines. The method illustrated in Figure 5.4 was used to do this.
2. Transformed the pixel coordinates of the points on the two intersecting lines to undistorted normalised image coordinates using Equations 2.18

to 2.21. This converted the coordinates into a rectilinear coordinate system. The intrinsic parameters and distortion coefficient for the optimal nonlinear calibration models in Table 3.3 were used to do this.

3. Calculated a simple linear regression for each line using the undistorted normalised image coordinates to give line equation $Y = mX + c$.
4. Found the intersection between the two lines using:

$$X = \frac{c_j - c_i}{m_i - m_j} \tag{6.1}$$

$$Y = m_i X + c_i \tag{6.2}$$

where $Y = m_i X + c_i$ and $Y = m_j X + c_j$ are the equations of the two lines that intersected (as shown in Figure 6.1).

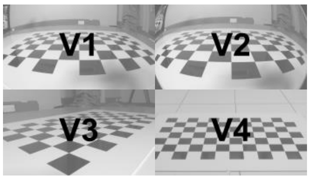
5. Transformed the intersection's coordinates from undistorted normalised image coordinates to pixel coordinates using Equations 2.11 to 2.17.
6. Calculated the distance, in the U and V axes, between the coordinates from step 5 and the sub-pixel accurate coordinates for Ci (i = 1 to 4) that were found in Chapter 3. This was called the line-line error.
7. Calculated the standard deviation of the Euclidean distance between the coordinates from step 5 and the sub-pixel accurate coordinates for Ci (i = 1 to 4) that were found in Chapter 3. This was called the line-line precision.

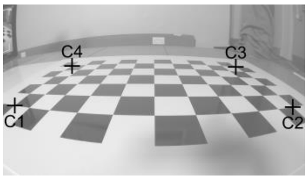
6.3 Results

The mean ± standard deviation of the line-line error for the four control points Ci (i = 1 to 4) in all viewpoints is shown in Table 6.1. The line-line error in the V axis was similar for all control points and smaller than it was in the U axis.

Table 6.1. Mean ± standard deviation of the line-line error in the U, V, X and Y axes. n = 10000 for each control point.

Viewpoint	Control point	U axis (pixels)	V axis (pixels)	X axis (10^{-3} m)	Y axis (10^{-3} m)
V1	C1	0.00 ± 0.13	0.00 ± 0.07	-0.2 ± 8.7	0.0 ± 3.5
	C2	0.00 ± 0.13	0.00 ± 0.07	0.2 ± 8.9	0.1 ± 3.5
	C3	0.00 ± 0.21	0.00 ± 0.08	-0.1 ± 12.4	0.2 ± 14.7
	C4	0.00 ± 0.21	0.00 ± 0.08	0.2 ± 11.9	0.1 ± 14.7
V2	C1	0.00 ± 0.11	0.00 ± 0.07	-0.2 ± 10.4	0.0 ± 3.5
	C2	0.00 ± 0.11	0.00 ± 0.07	0.1 ± 10.5	0.1 ± 3.5
	C3	0.00 ± 0.22	0.00 ± 0.08	-0.1 ± 12.6	0.2 ± 13.8
	C4	0.00 ± 0.21	0.00 ± 0.08	0.2 ± 12.3	0.0 ± 13.8
V3	C1	0.00 ± 0.14	0.00 ± 0.09	0.0 ± 2.8	0.0 ± 2.5
	C2	0.00 ± 0.19	0.00 ± 0.09	0.6 ± 23.5	0.0 ± 4.2
	C3	-0.01 ± 0.33	0.00 ± 0.07	-0.4 ± 27.1	0.2 ± 12.4
	C4	0.00 ± 0.18	0.00 ± 0.08	0.0 ± 3.8	0.0 ± 8.5
V4	C1	0.00 ± 0.11	0.00 ± 0.08	0.0 ± 3.3	0.0 ± 4.0
	C2	0.00 ± 0.11	0.00 ± 0.08	0.0 ± 3.3	0.1 ± 4.0
	C3	0.00 ± 0.10	0.00 ± 0.08	0.0 ± 3.9	0.1 ± 6.6
	C4	0.00 ± 0.10	0.00 ± 0.08	0.1 ± 3.9	0.0 ± 6.7





The line-line precision was 0.11 pixels, 0.11 pixels, 0.14 pixels and 0.07 pixels for V1, V2, V3 and V4 (respectively).

6.4 Discussion

Calculating control point pixel coordinates using the line-line method led to an increase in precision for all viewpoints. The repeatability study conducted in Chapter 5 showed that the existing method, i.e. digitising control points, could be done to a precision of 0.19 pixels. The line-line precision was between 0.07 and 0.14 pixels, which was an improvement of between 26% and 63%.

A key factor in this improvement was the use of an error minimisation technique. The use of more than two points to define a line provided extra spatial information about the line. Simple linear regression used this to minimise the effect that uncertainty in the points' coordinates had on the line calculated from them and hence the line-line intersections.

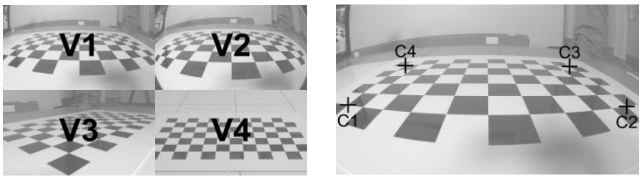
The line-line error in the U axis was larger than it was in the V axis. This was due to there being fewer points on the lines at 0 m and 50 m in the X axis than there were on those at 0 m and 25 m in the Y axis. As a result, for the lines at 0 m and 50 m in the X axis there was less spatial information to use when minimising the effect of uncertainty, which led to greater variability in an intersection's U coordinates. The disparity in point count was caused by the scale swimming pool used in this study. For a real swimming pool, the number of points could be the same on all lines and this should lessen this effect.

V3 was the non-perpendicular and non-fisheye viewpoint. C3 was the control point furthest from the camera.

The line-line method gave the smallest increase in precision for V3. For this viewpoint, C3 had the highest line-line error in the U axis. In the undistorted normalised image coordinate system, the angle between the two lines used to calculate C3 was the most obtuse. Table 6.2 has the angles between the lines in the undistorted normalised image coordinate system for all control points and viewpoints.

Table 6.2. Angle between the two lines used to calculate the line-line intersection at control points, Ci (i = 1 to 4). These angles were calculated in the undistorted normalised image coordinate system.

Viewpoint	Angle (degrees)			
	C1	C2	C3	C4
V1	23	23	157	157
V2	23	23	157	157
V3	134	30	163	33
V4	63	62	117	118



Consider the case shown in Figure 6.3. Line B had no error, but line A did. The error in line A, i.e. e_1, moved line A to be line A'. Because the two lines intersected at 90°, the error in the intersection between the two lines was e_1.

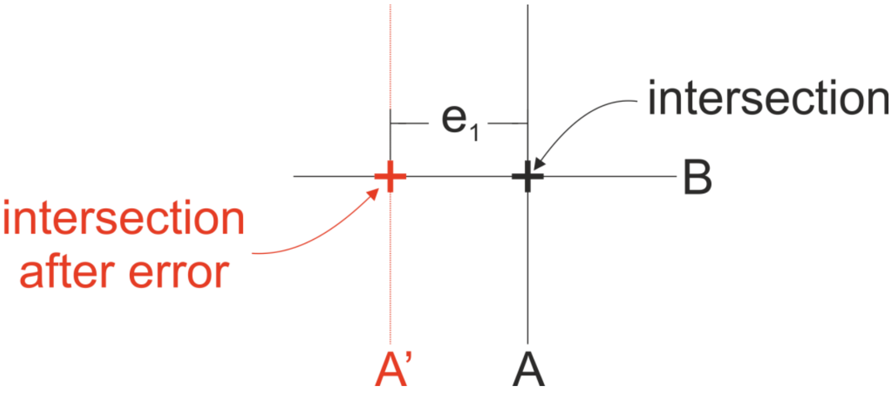


Figure 6.3. Effect of error on a line-line intersection when angle between lines was 90°.

If the angle between the two lines was not 90°, then the error in the intersection between the two lines would be greater than e_1. This is shown in Figure 6.4. The error in the intersection, i.e. e_2, was the length of the hypotenuse of the triangle between the two lines and the error in the line.

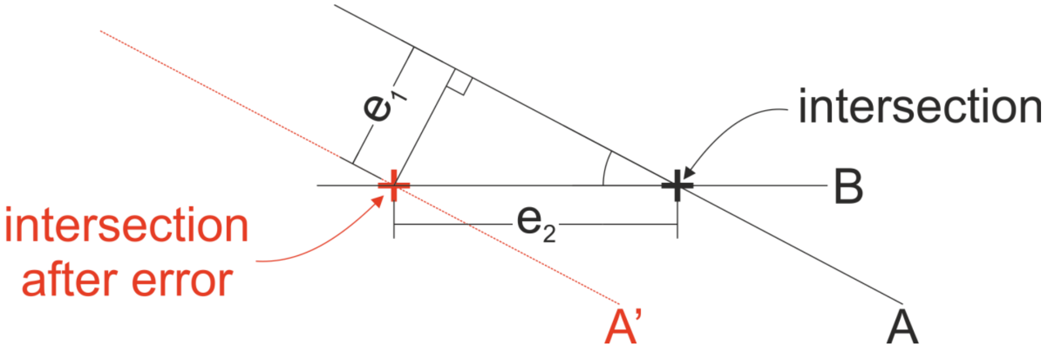


Figure 6.4. Effect of error on a line-line intersection when angle between lines was not 90°.

The angle between the intersecting lines at C3 for V3 was 163˚. Figure 6.5 shows that at this angle e_2 was almost four times greater than e_1. As a result, the low precision for C3 in V3 was partially caused by the highly obtuse angle between the lines used to calculate the intersection.

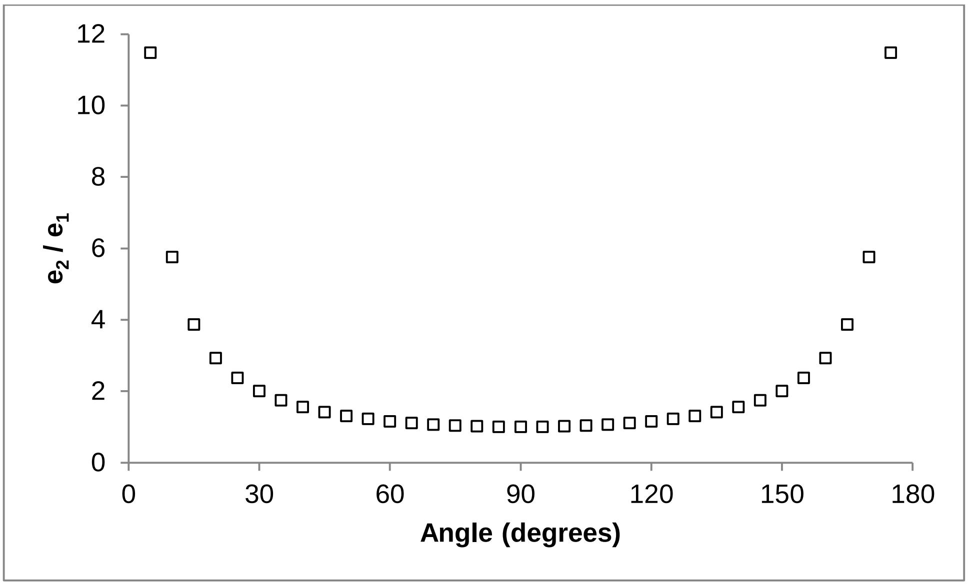


Figure 6.5. Relationship of the ratio of e_2 to e_1 and the angle between intersecting lines.

The mean of the line-line error in the U and V axes was small. It was zero pixels for all but one control point in one viewpoint. Line-line error was the distance between a control point's pixels coordinates when calculated by the line-line method and the sub-pixel accurate coordinates that were found in Chapter 3. So, the line-line method had a near-zero bias.

It is noteworthy that the line-line method would not be possible with the 2D-DLT calibration procedure. This is because the only rectilinear coordinate system in the 2D-DLT procedure is the world coordinate system and the transformation to this coordinate system requires control point pixel coordinates, which are calculated by the line-line method.

In practise the line-line method would require the digitisation of two or more points on each of the swimming pool's four bounding edges. A factor in how easy this would be is whether the edges could be clearly identified in the video footage. Anecdotally, as shown in Figure 6.6, some edges are well-defined whilst others can be obscured (by, in this case, advertising hoardings).

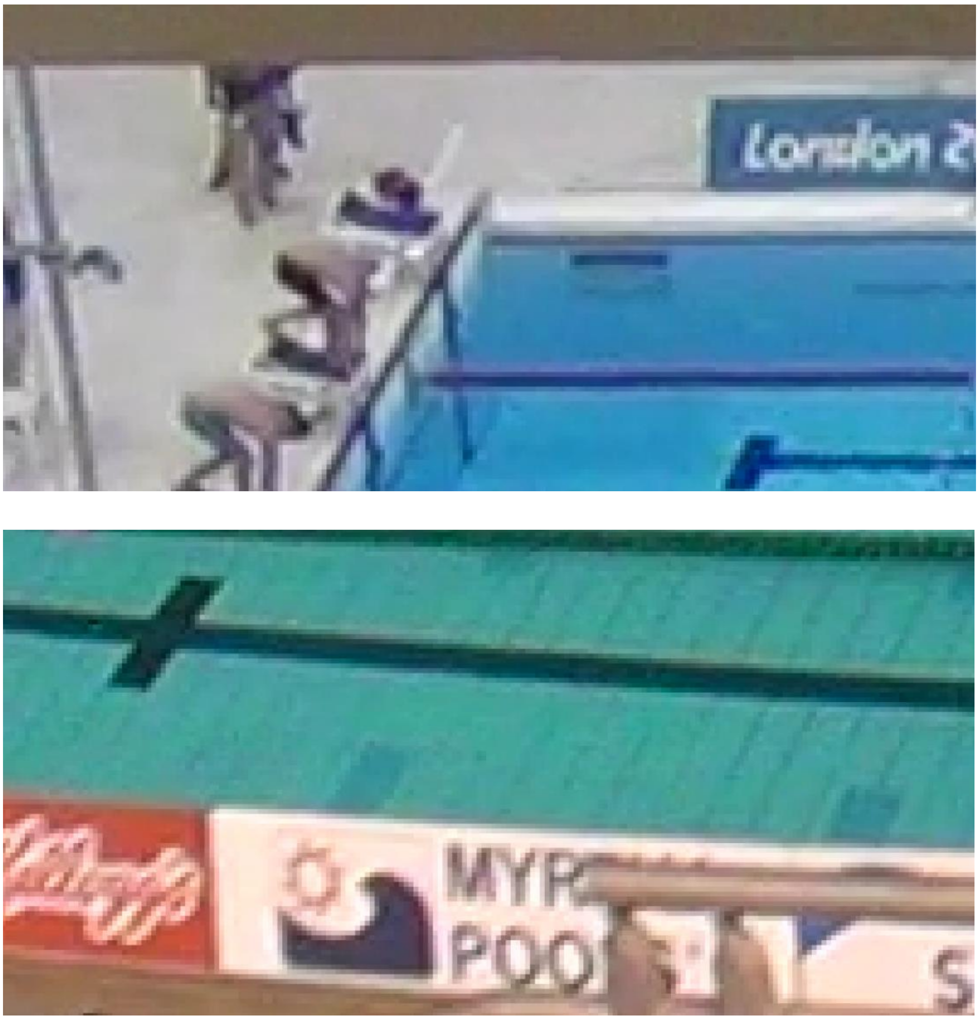


Figure 6.6. Example of swimming pool edges from competition analysis video footage.

6.5 Conclusion

This chapter presented a new method for determining a control point's pixel coordinates and found:

- For all viewpoints, the line-line method improved the precision with which a control point's pixel coordinates could be calculated. Improvements of between 26% and 63% were found.
- The line-line method should probably be used in preference to the current method of digitising a control point.
- The line-line method gave larger improvements for the perpendicular viewpoints than it did for the non-perpendicular one. So, improvements in control point error should be greater for the perpendicular viewpoints than they are for the non-perpendicular one.

In the next chapter, the effect that the line-line method has on control point error is quantified. This will confirm whether the line-line method should be preferred over the current method.

7 Control point error resulting from improved precision in a control point's pixel coordinates

7.1 Introduction

This chapter quantified the effect that the line-line method developed in Chapter 6 had on control point error. This was the error in reconstructing a distance that was caused by uncertainty in the measurement of the control points used to calculate a calibration model. Figure 7.1 shows its place in the hierarchy of errors; all the other errors in this hierarchy were fixed at zero in this chapter.

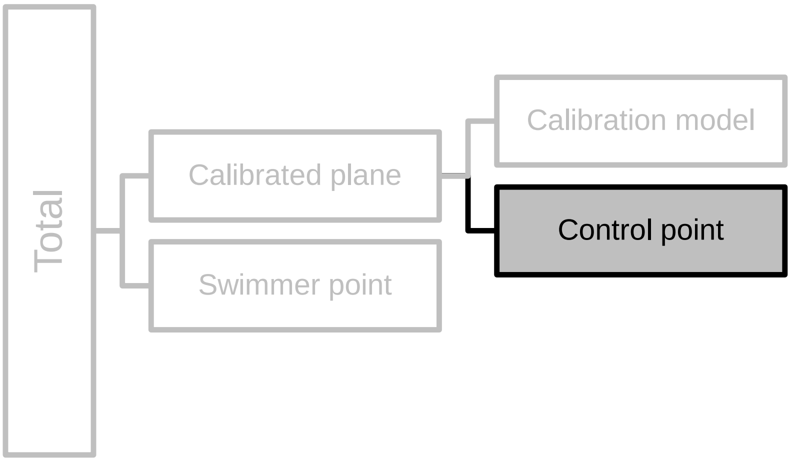


Figure 7.1. Chapter 7 re-calculated control point error. Calibration model and swimmer point errors were fixed at zero.

Control point error was calculated using a revised version of the computer simulation first described in Chapter 5. The simulated uncertainty in a control point's pixel coordinates was set by the results from the line-line method in Chapter 6.

7.2 Method

The study described in Chapter 5 was repeated, with the following modifications:

- Pixel coordinate system uncertainty. In Chapter 5, one distribution was used. In this chapter, one distribution per viewpoint was used. Each was a Gaussian normal distribution with a mean of zero pixels and a standard deviation given by the line-line precision from Chapter 6, i.e. 0.11 pixels, 0.11 pixels, 0.14 pixels and 0.07 pixels for V1, V2, V3 and V4 (respectively).
- Uncertainty was not added to the world coordinates of the control points.

One aspect not modified were the set of pixel coordinate error angles used in Chapter 5 (i.e. α in Figure 5.4). Using the same values in this chapter meant that only the magnitude of the uncertainty (i.e. ε in Figure 5.4) was changed. This isolated the effect on control point error of the improved precision given by the line-line method.

7.3 Results

The mean ± standard deviation of the control point error is shown in Table 7.1. Values for the current method of digitising a control point and the line-line method are given. Mean values were unchanged, but the standard deviation was reduced.

Table 7.1. Mean ± standard deviation of the control point error for the current and line-line method of calculating a control point's pixel coordinates.

Method	n	V1	V2	V3	V4
Current (10^{-3} m)	560000	0.0 ± 4.6	0.0 ± 4.5	0.0 ± 2.8	0.0 ± 0.7
Line-line (10^{-3} m)	560000	0.0 ± 2.7	0.0 ± 2.6	0.0 ± 2.1	0.0 ± 0.3

For V1, V2 and V4 the highest control point errors were in the outermost test distances. For V3 the highest were in the test distances that were furthest from the camera. This is shown in Figure 7.2.

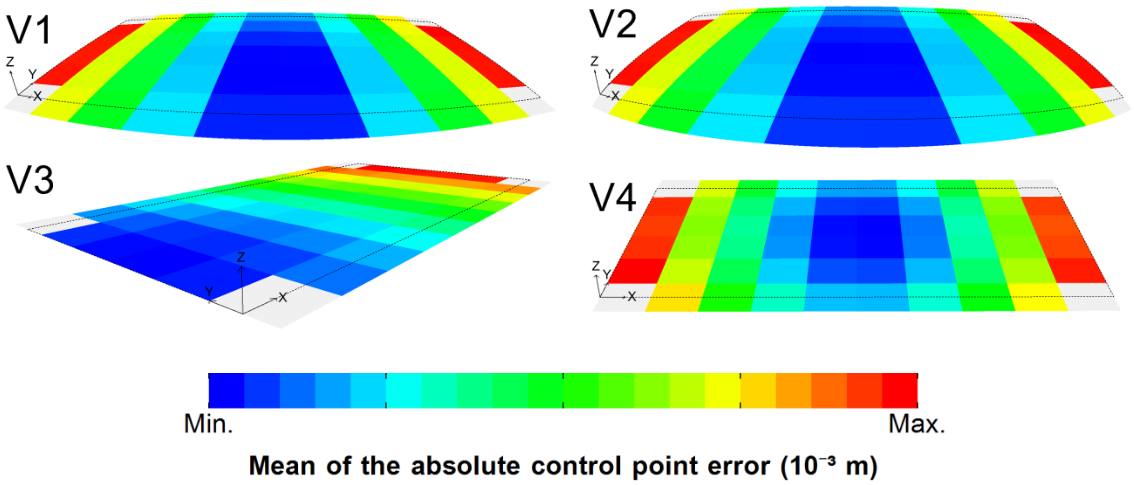


Figure 7.2. Pattern of the mean of the absolute control point error. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed. n = 10000 for each of the 56 squares in each heat map. Each heat map had its own scale with blue representing the viewpoint's minimum error and red the maximum.

7.4 Discussion

Overall, the improvement in the precision with which a control point's pixel coordinates could be calculated led to a reduction in control point error. The standard deviation of the control point error fell by 27% to 63%. So, the line-line method led to a reduction in control point error.

V1, V2 and V4 were perpendicular viewpoints and V3 was a non-perpendicular viewpoint.

The pattern of the absolute control point error was radial for V1, V2 and V4. The average control point error increased with distance from the centre of the calibrated plane. The uncertainties in the control points had the largest impact on the test distances that were closest to them. So, the largest RDEs in V1, V2 and V4 would be in the first and last 5 m of each lap. A similar radial pattern was not as clear for V3. Resolution was a key factor in this: the shorter test distances in pixels had higher average control point error. So, the largest RDEs in V3 would be between 45 m and 50 m in the X axis.

It is noteworthy that the largest RDEs were, in general, in the region of the calibrated plane occupied by the start, turn and finish phases. As a result, measurement of distances for the clean swimming phase would be less affected by uncertainty in a control point's pixels coordinates than would measurements of the other phases.

7.5 Conclusion

This chapter calculated control point error that resulted from using the line-line method to calculate a control point's pixel coordinates and found:

- The line-line method led to an improvement in control point error of between 27% and 63%.
- The line-line method should be preferred over the current method of digitising a control point.
- Control point error was likely to have a smaller impact on the measurements of distance in the clean swimming phase than it would in the other phases.

In the next chapter the combined effects of calibration model error (described in Chapter 3) and control point error (from this chapter and Chapter 5) are assessed. This gave the error in reconstructed distances that was due to the calibrated plane.

8 Calibrated plane error

8.1 Introduction

This chapter calculated calibrated plane error. This was the combination of calibration model and control point errors. It showed how sensitive a calibration model was to uncertainties in the measurement of the control points that it was calculated from. Figure 8.1 shows its place in the hierarchy of errors; swimmer point error was fixed at zero in this chapter.

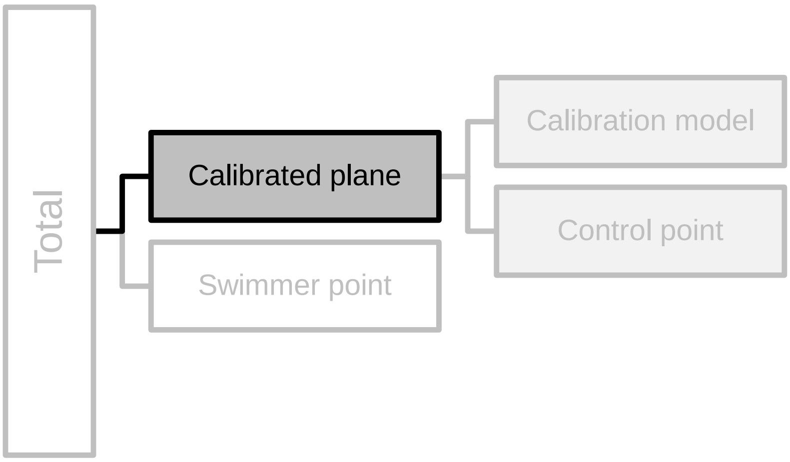


Figure 8.1. Chapter 8 calculated calibrated plane error. This was a combination of calibration model and control point errors. Swimmer point error was fixed at zero.

Calibrated plane error was calculated using a revised version of the computer simulation first described in Chapter 5. The main change to this simulation was the inclusion of calibration model error, which meant that RDE was a combination of calibration model and control point errors.

8.2 Method

The study described in Chapter 5 was repeated, with the following modifications:

- Inclusion of calibration model error. When calculating control point error in Chapters 5 and 7 the test distances had projected pixel coordinates. This removed calibration model error. In this chapter, the test distances had the pixel coordinates that were used when calculating calibration model error in Chapter 3.
- Pixel coordinate system uncertainty. In Chapter 5, one distribution was used. In this chapter, one distribution per viewpoint was used. Each was a Gaussian normal distribution with a mean of zero pixels and a standard deviation given by the results from Chapter 6, i.e. 0.11 pixels, 0.11 pixels, 0.14 pixels and 0.07 pixels for V1, V2, V3 and V4 (respectively).
- Uncertainty was added to both the pixel and world coordinates of the control points.
- RDE is called calibrated plane error in this chapter.

8.3 Results

The mean ± standard deviation of the calibrated plane error and the errors that were combined to make it are in Table 8.1. The values for calibration model error and calibrated plane error were similar.

Table 8.1. Mean ± standard deviation of the calibration model error, control point errors (for uncertainty in world and pixel coordinates of control points), and calibrated plane error.

Error	n	V1	V2	V3	V4
Calibration model (10^{-3} m)	56	10.0 ± 12.6	5.7 ± 19.8	-6.3 ± 39.1	0.0 ± 4.7
Control point world (10^{-3} m)	560000	0.2 ± 0.3	0.3 ± 0.3	0.3 ± 0.3	0.3 ± 0.2
Control point pixel (10^{-3} m)	560000	0.0 ± 2.7	0.0 ± 2.6	0.0 ± 2.1	0.0 ± 0.3
Calibrated plane (10^{-3} m)	560000	10.3 ± 12.8	6.0 ± 19.7	-6.0 ± 38.8	0.3 ± 4.6

The pattern of the mean of the absolute calibrated plane error in Figure 8.2 was similar to that for calibration model error shown in Figure 3.7.

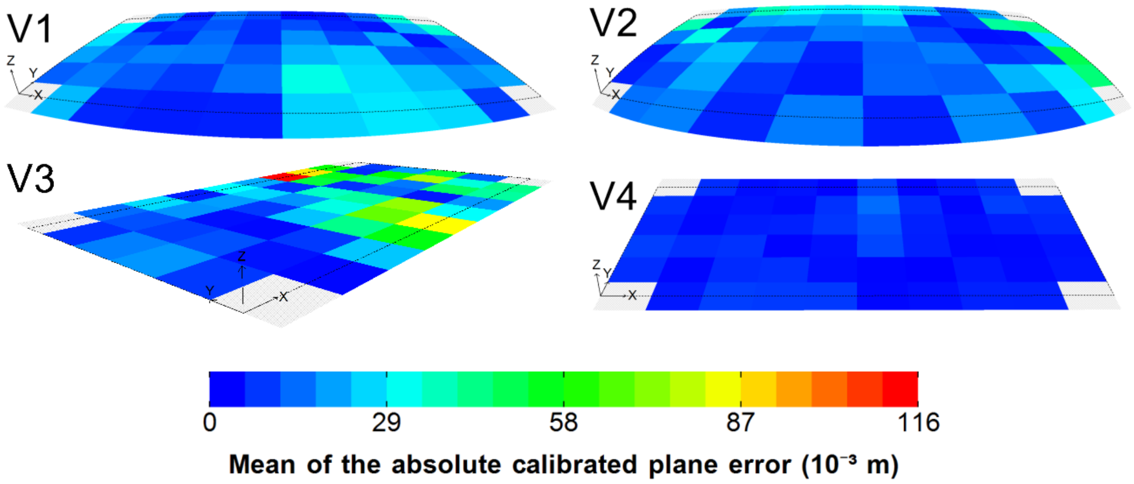


Figure 8.2. Pattern of the mean of the absolute calibrated plane error. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed. n = 10000 for each of the 56 squares in each heat map.

8.4 Discussion

The results showed that the nonlinear calibration models were not sensitive to uncertainties in the control points they were calculated from. These results supported the findings from Chapters 5 and 7, i.e. that a low-cost laser distance measurement device and the line-line method are appropriate methods of measuring control points.

8.5 Conclusion

This chapter calculated calibrated plane error and found:

- Nonlinear calibration models are not sensitive to uncertainties in the control points they were calculated from.

- If the swimming competition venue allowed it, the best viewpoint was perpendicular and non-fisheye. This viewpoint had the lowest calibrated plane error.
- A perpendicular and fisheye viewpoint should be chosen over a non-perpendicular and non-fisheye one.

This chapter marked the end of the assessment of the error in a reconstructed distance that was due to the calibrated plane. In the next chapter a calibrated plane was used to calculate the speed of swimmers in a competition.

9 Clean swimming speed from manual digitisation

9.1 Introduction

The use of the nonlinear calibration procedure and a single-camera viewpoint were assessed in Chapters 3 to 8. This chapter used these to calculate speed in the clean swimming phase at a major national competition. This was the British Gas Swimming Championships 2013; Figure 9.1 shows the filming position at this competition.

Figure 9.1. Filming position at the British Gas Swimming Championships 2013.

The races in one session were filmed and manually digitised. This session had a mix of races: men and women, all four competitive swimming strokes and sprint and middle distance races. The digitised points were reconstructed and used to calculate a swimmer’s speed in the clean swimming phase. This demonstrated the use of a nonlinear calibration procedure and a single-camera viewpoint for competition performance analysis. A sample of data that resulted from this is presented.

9.2 Method

With institutional ethics approval the racing at the British Gas Swimming Championships 2013 was filmed. The competition was held at the Ponds Forge International Sports Centre. It was the national championships, and for British swimmers, the main qualifying competition for the 2013 FINA World Championships. As a result, the best British swimmers competed.

9.2.1 Races

The finals in the last session of the competition, which was on the 30th July 2013, were filmed. This sample had at least one race per stroke, a mixture of distances and both sexes. A total of 206 laps swum by 71 swimmers were filmed. The races are summarised in Table 9.1.

Table 9.1. Races and number of laps used in the manual tracking study. The swimmer in lane 8 of the women's 100 m freestyle race did not start.

Stroke	Men	Women	Laps
Backstroke	200 m	50 m	40
Breaststroke	100 m	200 m	48
Butterfly	100 m	50 m	24
Freestyle	100 m	100 m	30
Individual Medley	400 m	-	64

9.2.2 Filming

A Sony HDR-PJ260VE camcorder fitted with a 0.42 multiplier fisheye lens converter was used to film the racing. The camera was located at approximately 15 m from the near edge and 10 m above the pool's surface. A staircase prevented the camera being positioned at the preferred central location. So, a location at 27 m in the X axis was used. The central eight of ten swimmer lanes were used. Lane 1 was closest to the camera and lane 8 was furthest from it. Races of 100 m and over started at 0 m in the X axis whereas the 50 m races started at 50 m in the X axis. Start strobes, which lit when a race was started, were placed in the camera's view. The camera location and pool layout are shown in Figure 9.2.

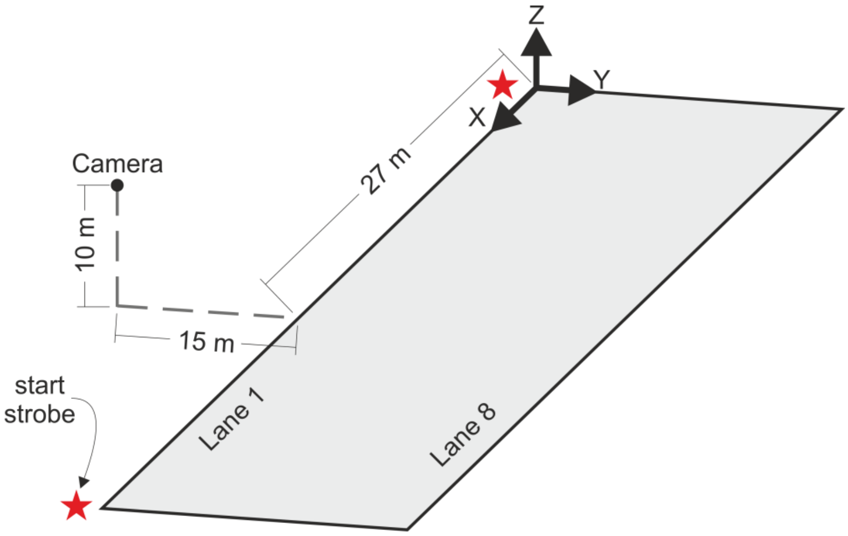


Figure 9.2. Camera location and layout of the pool at the British Gas Swimming Championships 2013.

The camcorder recorded at a frame rate of 50 Hz and captured FHD footage, i.e. images that were 1920 pixels wide by 1080 pixels high. The footage was recorded in the Advanced Video Codec High Definition format and stored in an .MTS file. After the competition, frames were extracted from the video file and stored as Portable Network Graphic (PNG) image files. The filename of each was set using the frame number, e.g. frame 100 was stored in file 000100.PNG.

9.2.3 Official results

Data from the competition's timing and results system, which included lane allocations, split times (i.e. time at the end of each lap) and finishing positions (i.e. first, second, third, etc.), were downloaded from the British Swimming website (British Swimming 2014).

9.2.4 Digitisation

The swimmers were manually digitised in a custom software coordinate digitiser (a screenshot of which is in Figure 9.3). For each race, split times and video time were linked by identifying the frame in which the start strobes first lit: this was 0 s in video time. Then, the footage of the race was replayed frame-by-frame and the swimmers in each lane were digitised.

Frame	U	V
8141	1310.66	603.59

8141 1310.53 604.09 8

Figure 9.3. Screenshot of the custom software coordinate digitiser.

The centre of a swimmer’s head was digitised using a variable diameter circular digitisation cursor. First, the footage was zoomed to enlarge the view of the swimmer’s head; in the example shown in Figure 9.3 an eight times zoom was used. Next, the cursor’s diameter was adjusted to match that of the outline of the swimmer’s head. Then, the cursor was aligned with the outline of the swimmer’s head. Lastly, when the mouse was clicked, the pixel coordinates at the centre of the cursor’s circle were recorded.

A consideration when doing this was to ensure that the centre of a swimmer's head was on the calibrated plane when it was digitised. For backstroke and freestyle, digitisation began when the swimmer's head first emerged after the post-start or post-turn underwater swim. It ended on the last hand entry before the turn or end of the race. For breaststroke and butterfly, digitisation also began on the first head emersion after the post-start or post-turn underwater swim. It ended on the last head emersion before the turn or end of the race. This was the same approach as that used by Veiga et al. (2012). For backstroke and freestyle, a swimmer was digitised in every frame between the start and end actions. For breaststroke and butterfly a swimmer was only digitised when the centre of their head was on the calibrated plane as it moved upward in each stroke cycle; so, only one digitisation per stroke cycle was attempted for these strokes.

9.2.5 Reconstruction

A nonlinear calibration model was used to reconstruct the digitised coordinates of the swimmers. The nonlinear calibration model was calculated using the process described in Section 2.3. Control points were at the corners of the swimming pool at the water surface.

Reconstructed distance error (RDE) could not be assessed in the same way that it had been in Chapters 3, 5, 6, 7 and 8. This was because ground truth world and pixel coordinates of points on the calibrated plane were not known. So, another approach was sought. British Swimming attended the competition and constructed a lane rope map, as described in Section 2.3. This map calculated that each float was between 0.08 m and 0.09 m in the X axis. In addition, it identified the floats on the lane ropes that were at 5 m, 10 m, 15 m, 25 m, 35 m, 40 m and 45 m in the X axis; this is shown in Figure 9.4. The lane ropes were between a half and two floats out of alignment.

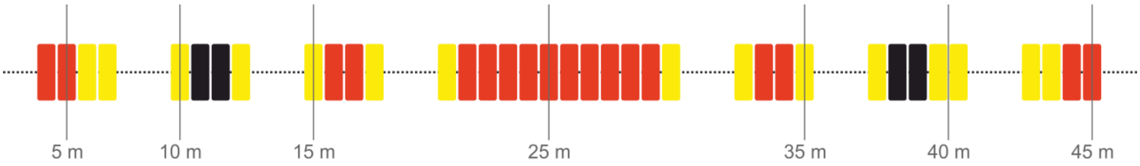


Figure 9.4. Lane rope floats at key distances in the X axis.

RDE was calculated for distances on the lane ropes. The floats at 5 m, 10 m, 15 m, 25 m, 35 m, 40 m and 45 m in the X axis on nine lane ropes were digitised. These were the lane ropes for the central eight lanes in the pool. The digitisations were reconstructed and RDE for distances between adjacent floats was calculated to give six RDE for each lane rope.

9.2.6 Swimmer speed

Speed per lap in the clean swimming phase, or *v*, was calculated from the reconstructed coordinates using simple linear regression: the reconstructed X coordinates were plotted against time in the race and the gradient of the linear trend line through the points gave *v*.

9.3 Results

The *v* for all laps in all races is in Appendix 15.2. A sample is repeated in Table 9.2: these are for the men’s and women’s 100 m freestyle races. The swimmers in lane 4 won the races, the woman in lane 7 on lap 1 could not be tracked, and the woman in lane 8 withdrew from the race.

Table 9.2. *v* in the men's and women's 100 m freestyle races.

Lane	Lap	Men v (m s^{-1})	Women v (m s^{-1})
1	1	2.037	1.821
	2	1.856	1.633
2	1	2.046	1.802
	2	1.875	1.692
3	1	2.051	1.835
	2	1.937	1.700
4	1	2.114	1.813
	2	1.903	1.737
5	1	2.016	1.802
	2	1.941	1.736
6	1	2.002	1.824
	2	1.879	1.686
7	1	1.990	-
	2	1.867	1.717
8	1	2.010	-
	2	1.847	-

The mean ± standard deviation of the time to digitise a swimmer in a lap is shown in Table 9.3. On average, it took two to three times longer to digitise a lap from a backstroke or freestyle race than it did for the other two strokes. The total time taken was 29 hours, 11 minutes and 6 seconds.

Table 9.3. Mean ± standard deviation of the time to digitise a swimmer in a lap.

Stroke	n	Time (min:sec)
Backstroke	56	12:49 ± 3:53
Breaststroke	64	4:46 ± 1:34
Butterfly	40	5:27 ± 1:20
Freestyle	44	11:36 ± 2:57

The RMSE of the calibrated plane error was 85.3 x 10^{-3} m (n = 54).

9.4 Discussion

9.4.1 Digitisation

Indistinct swimming caps, splash, background objects and low resolution made choosing the frame to digitise and positioning the digitisation cursor problematic. Examples of these problems are in Figure 9.5.

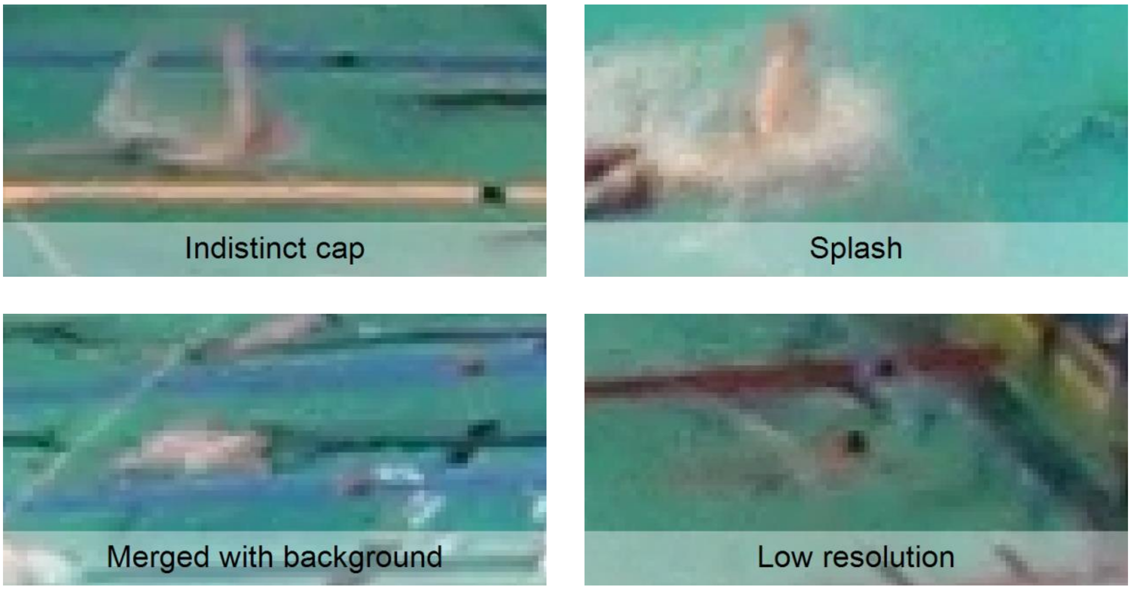


Figure 9.5. Problems encountered during digitisation. The swimmer's head was approximately central in each image.

An indistinct cap made it hard to identify the swimmer's head and hence size and position the circular digitisation cursor. Blue caps were similar in colour to the water and red and yellow ones were similar in colour to the swimmer's skin and the splash created by the swimmer. In addition, caps with detailing (e.g. a swimmer name, club name or a club logo) were harder to identify than a solid colour cap. Solid white caps were the easiest to identify and position the digitisation cursor around. An example of a white cap is in Figure 9.6.

Figure 9.6. Solid white swimming caps were the easiest to identify and position the digitisation cursor around.

It was found that the swimmer’s cap was quite often obscured by splash. There seemed to be more splash in the sprint races than there was in the middle distance races. There was typically less splash in the breaststroke races than there was for the other strokes. If the head was fully obscured by splash then the head was not digitised, but if it was only partially obscured then it was sometimes possible to align the circular digitisation cursor with the visible parts of the head and digitise the swimmer. This was the reason this type of cursor was used.

A further problem was that the swimmer’s head sometimes merged with, or was obscured by, a background object. In certain locations on the calibrated plane the swimmer’s head joined with the black line on the pool’s floor. This was caused by the relative position and orientation between camera, swimmer and background object. Another, more frequent, occurrence was the lane rope obscuring the view of a swimmer’s head. This was most common in backstroke when a swimmer swam beside the lane rope nearest to the camera. As with splash, it was sometimes possible to overcome this problem by aligning the circular digitisation cursor with the visible parts of the head.

The final problem was low resolution. As shown in Figure 4.4, resolution in the resultant direction for a perpendicular and fisheye viewpoint was lowest toward the start and turn ends and in the lanes most distant from the camera. In these regions, a swimmer's head, which was a fixed size in the world coordinate system, was smaller than elsewhere on the calibrated plane. This led to difficulties in choosing the frame to digitise for breaststroke and butterfly and uncertainty in where to position the circular digitisation cursor for all strokes.

A result of these problems was incomplete tracking. In most cases, some digitisations were missed in each lap. At worst no tracking was possible, which was the case for 2 of the 206 laps. In one of these laps, the swimmer was in a far lane, wore a red cap and swam adjacent to the lane rope nearest to the camera. In the other, the swimmer was also in a far lane and produced a large amount of splash. The cumulative effect of these problems prevented tracking. Overall, swimmers were successfully tracked in 204 of the 206 laps, which was a failure rate of less than 1%.

9.4.2 Reconstruction

It was not possible to definitively identify the source of the RDE. Likely sources were: British Swimming's measurements, movement of the lane ropes after measurement, digitisation of floats, calibration model error or control point error. It is probable that these all contributed. Nevertheless, the error was relatively small: it was about one lane rope float.

A further visual assessment of the calibrated plane was made. This involved undistorting an image of the swimming pool. The resultant image showed that lines which were straight in the world, e.g. edges of the pool and lane ropes, were also straight in the undistorted image. The undistorted image is shown in

Figure 9.7. This gave further confirmation that the calibrated plane was acceptable; critically, it showed that the radial distortion that resulted from using a fisheye lens converter had been successfully accounted for.

Figure 9.7. Undistorted image of the calibrated plane. Straight lines in the world, which were curved in the video footage, became straight in the undistorted image.

9.4.3 Case studies

One swimmer in two races was selected for in-depth analysis. One race was the women's 50 m butterfly and the other was the women's 100 m freestyle. Both involved the same swimmer, who won both races. The swimmer was in lane 3 for the butterfly race and lane 4 for the freestyle race. These cases were selected as the swimmer was one of the world's best: she had the 4th best race time in 50 m butterfly in 2013 (FINA 2015b).

Three plots were created for each lap. First, reconstructed X coordinate was plotted against time. A simple linear regression line was plotted on this to highlight the method used to calculate *v*. Secondly, the rate of change of the reconstructed X coordinate was calculated as:

$$\delta X_i = \frac{X_{i+k} - X_{i-k}}{t_{i+k} - t_{i-k}} \tag{9.1}$$

where X_{i+k} and X_{i-k} were the reconstructed X coordinate in frame $i + k$ and $i - k$ (respectively), t_{i+k} and t_{i-k} the times for these frames, and k the window size. For the butterfly race k was one; due to the proximity of adjacent digitisations in the freestyle race, k was set at twenty-five. An order two polynomial trend line was fitted to the δX data. The third, and final, plot was residual against distance. Residual was the difference between a reconstructed X coordinate and that predicted by the simple linear regression used to calculate *v*. An order two polynomial trend line was fitted to the residual data.

The plots for the butterfly case study are shown in Figure 9.8.

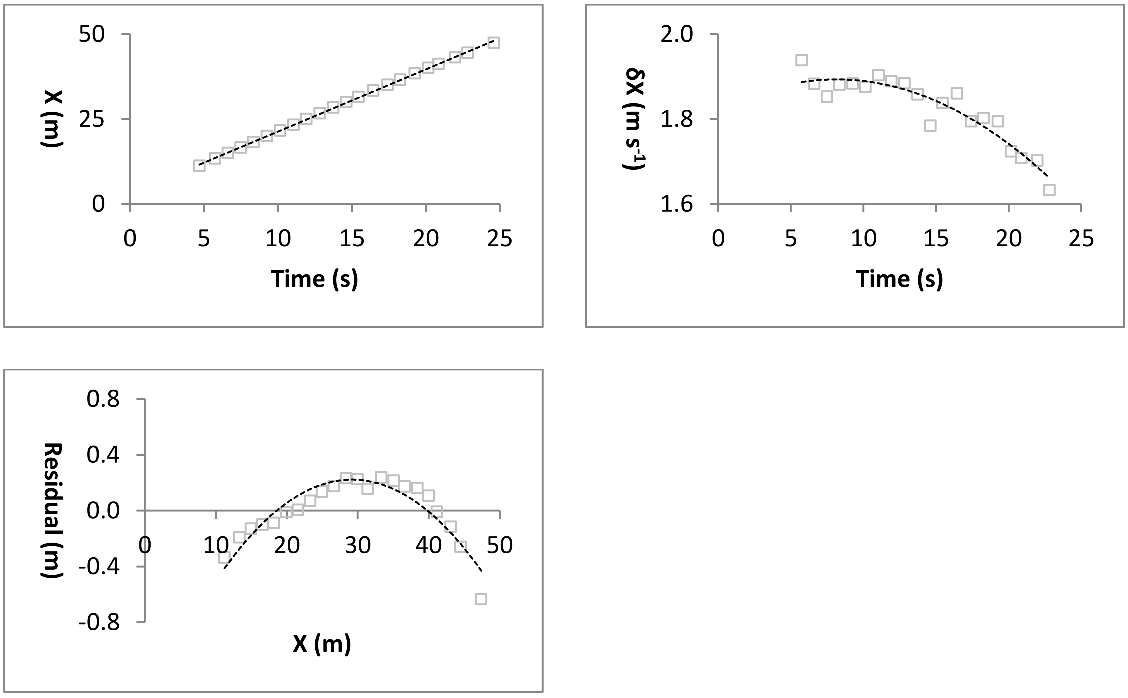


Figure 9.8. Women 50 m butterfly case study. The black dashed lines are trend lines through the points.

Distance against time formed an approximately straight line. The R^2 for the simple linear regression was 0.99964. The two other plots showed that the swimmer did not appear to maintain a constant speed throughout the clean swimming phase. The order two polynomial trend line for δX suggested a gradual decrease in speed as the race progressed, which caused the residual plot to have an inverted U shape. Residual in the final digitised frame was about 0.63 m. This distance was the equivalent of seven to eight lane rope floats. Given the checks of the calibrated plane described above, it was most likely that the swimmer did reduce speed toward the end of this lap. An alternative explanation was that the residual was due to error in the calibrated plane. An error of this size would have been obvious when the calibrated plane was checked.

A gradual decrease in speed as the race progressed fitted intuition: the sprinter started quickly and attempted, but failed, to maintain a high speed throughout the race. In addition, when she was approximately 40 m (or 20 s) into the race, she was at least half a body length clear of the other swimmers. As such, she may have decided to reduce her speed at this distance.

The plots for the freestyle case study are shown in Figure 9.9.

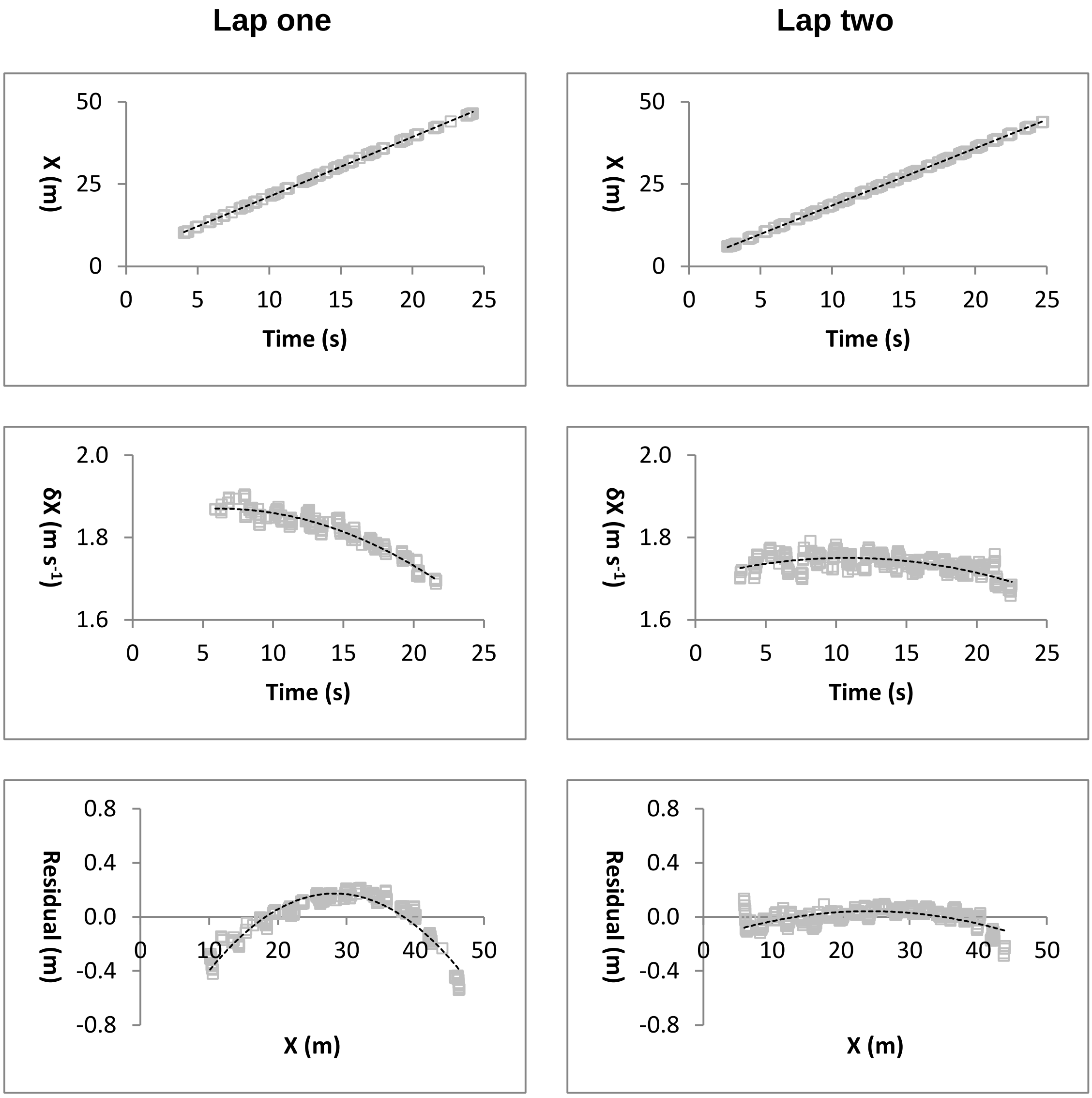


Figure 9.9. Women’s 100 m freestyle case study. The black dashed lines are trend lines through the points.

The freestyle case study had similarities with the butterfly one: distance against time plots were approximately straight and had high R^2 of 0.99964 and 0.99996 for laps 1 and 2 (respectively) and the residual had an inverted U shape. There were clear differences between laps 1 and 2: lap 1 was similar to the butterfly case study but lap 2 was not. In lap 2 the swimmer’s initial δX was lower than it was in lap 1 and there was lower variability in this measure throughout the lap. This was reflected in the residual, which showed that the swimmer remained in close proximity to the trend line throughout lap 2. An interpretation of this is that the swimmer gave a near maximal effort in the first lap and then, whilst fatigued, attempted to maintain her highest v in the second lap. An additional factor may have been that the start phase contributed to a high initial δX in lap 1, but clearly not lap 2.

These case studies illustrated some advantages of using a single-camera viewpoint and digitising the swimmer as often as possible. First, an insight into speed within a lap was gained. Contrast this with the competition performance analysis reported in the literature: the swimmer is digitised twice, i.e. once at the start and once at the end of the clean swimming phase and so such insight is not possible. Secondly, an observation of the case studies suggested that calculating speed using simple linear regression helped to minimise the effect on the calculated speed of uncertainty in swimmer digitisations. This observation is tested in Chapter 11.

These advantages come at a cost. Each swimmer had to be digitised more than twice in each lap. This took about five to twelve minutes per lap. How this compares to the methods reported in the literature is not clear, as digitisation time is typically not reported. It may be that the times found in this study are typical for competition performance analysis. Indeed, it is likely that digitisation time for a lap would take minutes rather than seconds irrespective of the method used.

9.5 Conclusion

This chapter calculated clean swimming speed at a major national competition and found:

- Digitising the swimmer was not always straightforward. In particular, indistinct swimming caps, low resolution, splash and background objects made choosing the frame to digitise and positioning the digitisation cursor problematic.
- Swimmers may not maintain a constant speed throughout the clean swimming phase. This is a new finding that was made possible by the use of a nonlinear calibration procedure and a single-camera viewpoint.

The next chapter looks at the impact of uncertainty in swimmer digitisations. It shows how the digitisation problems identified in this chapter could propagate to errors in reconstructed distances and hence calculation of speed.

10 Swimmer point error

10.1 Introduction

This chapter calculated swimmer point error. This was the error in reconstructing a distance that was caused by uncertainty in the digitisation of a swimmer. Figure 10.1 shows its place in the hierarchy of errors; all the other errors in this hierarchy were fixed at zero in this chapter.

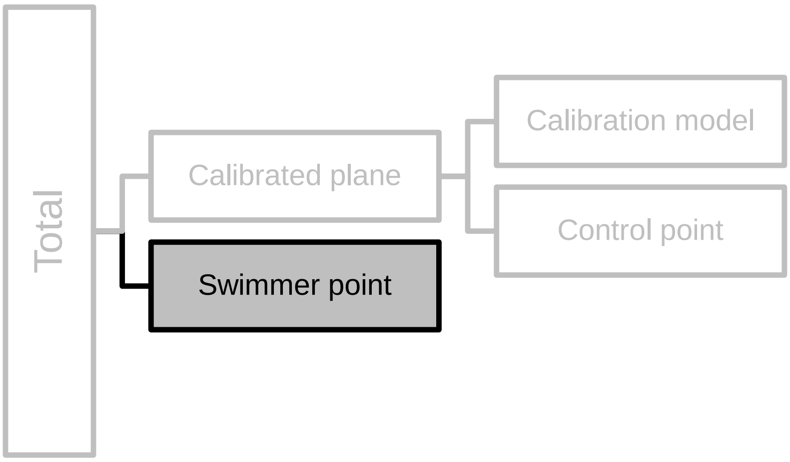


Figure 10.1. Chapter 10 calculated swimmer point error. Calibration model and control point errors were fixed at zero.

The previous chapter identified digitisation problems. For example, the swimmer's head was sometimes obscured by splash or a lane rope. Veiga et al. (2010) recognised that such problems could lead to uncertainty in digitisations. So, they did an intra-analyst precision study to quantify this. One analyst repeatedly digitised the start and end points of a turn in each of eight swimming lanes. Thirty repeats were performed for each lane. The standard deviations of the turn distances were between 15.5 x 10^{-3} m and 34.3 x 10^{-3} m for all eight lanes. In general, they found that the standard deviation increased as the distance from the camera did.

The work in this chapter starts with a study similar to the one done by Veiga et al. (2010). The data from this is then used to assess the effect that uncertainty in digitisation of a swimmer would have on error in reconstructed distances. A revised version of the computer simulation first described in Chapter 5 was used to do this.

10.2 Intra-analyst precision

10.2.1 Method

An assessment of precision in swimmer digitisation was carried out. A key consideration in the study's design was ensuring that the four competitive swimming strokes and the problems identified in Chapter 9 (e.g. indistinct cap colour, low resolution) were considered.

The men's 400 individual medley race from the British Gas Swimming Championships 2013 was selected. This gave the four swimming strokes. Also, the swimmers in this race used a variety of cap colours including those that proved easy (i.e. solid white) and difficult (i.e. red with lettering and a logo) to isolate from the background. Furthermore, this race had examples of swimmers in close proximity to the lane ropes, which often led to problems when digitising.

Each of the eight lanes was split into three segments in the X axis: 0 m to 15 m, 15 m to 35 m and 35 m to 50 m. One swimmer point per stroke for each of these 24 segments was then selected, at random, from those digitised in Chapter 9. This gave a total of 96 swimmer points and ensured that there were points in every lane, including the lower resolution segments of each lane. The selected points are shown in Figure 10.2.

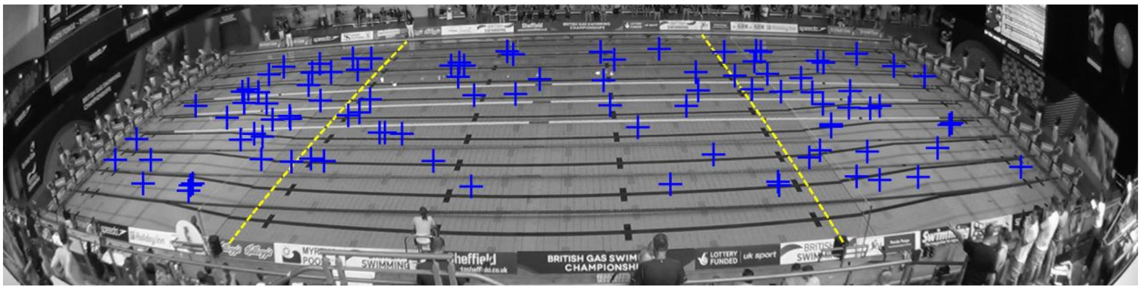


Figure 10.2. The 96 randomly selected swimmer points that were used in the digitisation precision study.

The 96 swimmer points were digitised 20 times with at least 24 hours between repeats. A custom software coordinate digitiser was used; this was based on the one used in Chapter 9 and hence had the same features, e.g. a circular digitisation marker. The mean U and V coordinate of the repeats of each swimmer point was taken as the best estimate of its ground truth pixel coordinates. The digitised points were randomly spread about the mean; this is shown for two of the swimmer points in Figure 10.3.

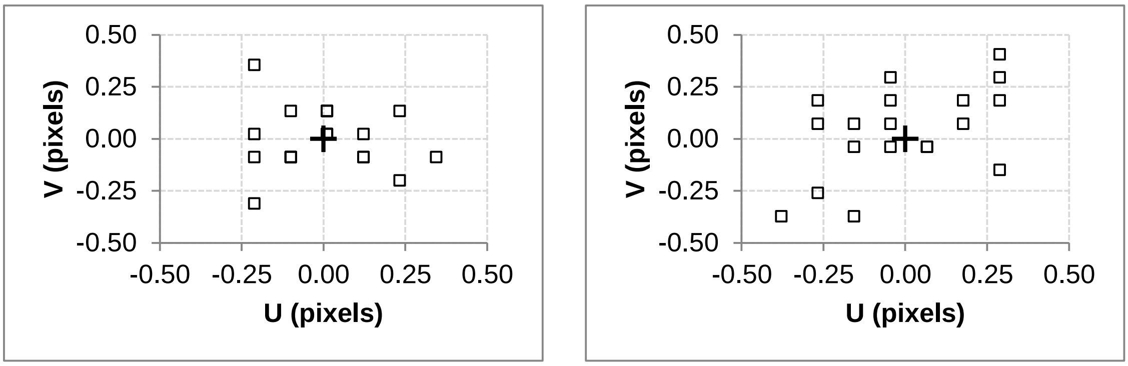


Figure 10.3. Visualisation of the repeated digitisation of two swimmer points. The cross is the mean of the digitisations. A square shows the difference between an individual digitisation and the mean.

The Euclidean distance between each digitised coordinate and the mean for the corresponding swimmer point was calculated. This was called digitised coordinate distance.

10.2.2 Results

The standard deviation of the digitised coordinate distance for all 96 swimmer points was 0.28 pixels to two decimal places. There was little correlation between the standard deviation of the digitised coordinate distance and lane number: Pearson's *r* was 0.063. In addition, Table 10.1 showed that the highest standard deviation of the digitised coordinate distance was in one of the nearest lanes whilst the lowest was in the second furthest lane.

Table 10.1. Standard deviation of the digitised coordinate distance in lanes 1 to 8.

Lane	n	Std. dev. (pixels)
1	240	0.21
2	240	0.22
3	240	0.38
4	240	0.27
5	240	0.26
6	240	0.32
7	240	0.17
8	240	0.30

Furthermore, there wasn't a consistent pattern of standard deviation of the digitised coordinate distance in the three segments that were used to assign swimmer points in each lane. This is shown in Figure 10.4.

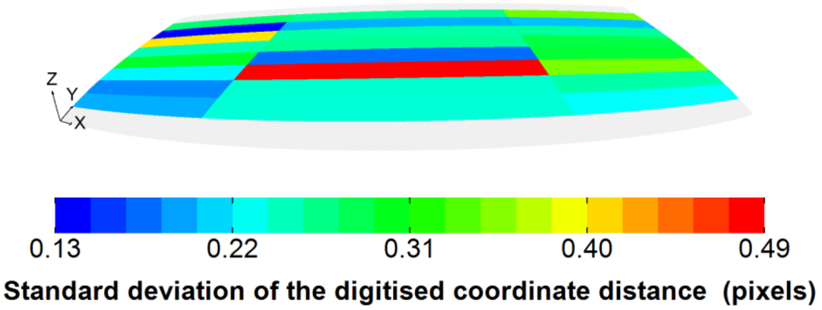


Figure 10.4. Pattern of the standard deviation of the digitised coordinate distance. There weren't any swimmer points in the grey hatched areas. n = 80 for each of the 3 segments per lane.

The effect that cap colour had on digitised coordinate distance was tested for lanes 6 and 7. The swimmer in lane 6 wore a red cap, which had a club logo and either a swimmer name or club name on it. The swimmer in lane 7 wore a solid white cap. As discussed in Chapter 9, the cap in lane 6 was amongst the hardest to isolate from the background whilst the one in lane 7 was one of the easiest. Cohen's (1992) *d* index was used to assess effect size:

$$d = \frac{\overline{DCD_r} - \overline{DCD_w}}{\sigma_{pooled}} \tag{10.1}$$

$$\sigma_{pooled} = \sqrt{\frac{\sigma_r^2 + \sigma_w^2}{2}} \tag{10.2}$$

where $\overline{DCD_r}$ and $\overline{DCD_w}$ are the mean of the digitised coordinate distance for red and white caps (respectively) and σ_r and σ_w are the standard deviation of the digitised coordinate distance for red and white caps (respectively). *d* was 0.63, which indicated that cap colour had a large effect on intra-analyst precision in these two adjacent lanes.

10.3 Simulation

10.3.1 Method

The effect that the intra-analyst precision could have on error in reconstructed distances was assessed by a revised version of the computer simulation described in Chapter 5. The following modifications were made:

- Uncertainty in the control points' world and pixel coordinates were fixed at zero.
- Uncertainty was added to a swimmer point's pixel coordinates using the method illustrated in Figure 5.4, i.e. a random angle in the range 0° to

359° and a random magnitude from a Gaussian normal distribution. This distribution had a mean of zero pixels and a standard deviation found by the intra-analyst precision study, i.e. 0.28 pixels.

- RDE was called swimmer point error in this chapter.

Ten thousand independent trials were done. Each trial added uncertainty to the test distance endpoints shown in Figure 3.6, reconstructed the test distances, and calculated their RDE.

10.3.2 Results

The standard deviation of the swimmer point error found by Veiga et al. (2010) is in Table 10.2. Analysis of their data showed a positive correlation between the standard deviation of the swimmer point error and lane number: Pearson's *r* was 0.851.

Table 10.2. Standard deviation of the swimmer point error from Veiga et al. (2010).

Lane	n	Std. dev. (10^{-3} m)
1	30	15.5
2	30	20.8
3	30	24.3
4	30	27.4
5	30	31.3
6	30	25.5
7	30	27.2
8	30	34.3

The mean and standard deviation of the swimmer point error from the computer simulation are shown in Table 10.3. The computer simulation did not use lane numbers, but lane number and test distance Y coordinate are directly related (as shown in Figure 3.5). A positive correlation between the standard deviation of the swimmer point error and test distance Y coordinate was found: Pearson's *r* was 0.931, 0.899, 0.956 and 0.997 for V1 to V4 (respectively).

Table 10.3. Mean ± standard deviation of the swimmer point error for test distances at 0 m to 25 m in the Y axis.

Y (m)	n	V1 (10^{-3} m)	V2 (10^{-3} m)	V3 (10^{-3} m)	V4 (10^{-3} m)
0	80000	0.0 ± 10.1	0.0 ± 10.2	0.0 ± 33.0	0.0 ± 8.3
5	100000	0.0 ± 14.6	0.0 ± 14.9	0.0 ± 37.8	0.0 ± 8.9
10	100000	0.0 ± 16.8	0.0 ± 16.5	0.0 ± 40.0	0.0 ± 9.3
15	100000	0.0 ± 18.9	0.0 ± 18.3	0.0 ± 42.3	0.0 ± 9.8
20	100000	0.0 ± 21.2	0.0 ± 20.4	0.0 ± 44.1	0.0 ± 10.3
25	80000	0.0 ± 19.9	0.0 ± 18.7	0.0 ± 44.2	0.0 ± 10.6

In addition to increasing swimmer point error with increasing test distance Y coordinate, Figure 10.5 showed that for V1, V2 and V4 it increased with distance from the optical axis, i.e. 25 m in the X axis.

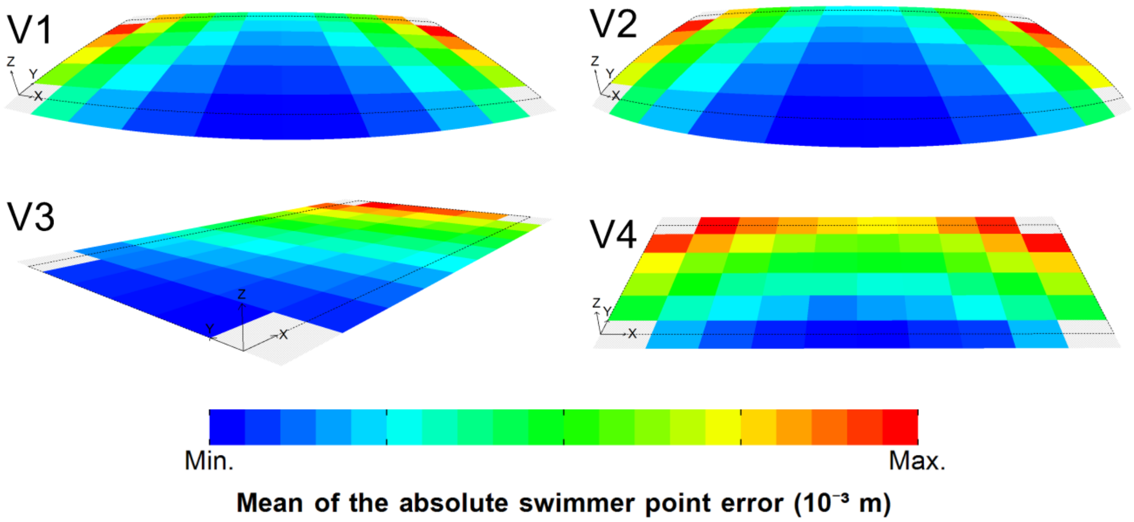


Figure 10.5. Pattern of the mean of the absolute swimmer point error. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed. n = 10000 for each of the 56 squares in each heat map. Each heat map had its own scale with blue representing the viewpoint's minimum error and red the maximum.

10.4 Discussion

10.4.1 Intra-analyst precision

Almost no correlation between intra-analyst swimmer digitisation precision and lane number was found. Results from Chapter 4 showed that the lowest resolutions were in the lanes furthest from the camera and, for a fisheye viewpoint similar to one used in this study, the first and last 15 m of each lane. However, as shown Figure 10.4, these regions of the calibrated plane did not exhibit a consistent pattern of low precision swimmer point digitisations. This suggested that resolution was not a central factor in intra-analyst swimmer digitisation precision.

Swimmer's cap colour was shown to have a large effect on intra-analyst swimmer digitisation precision. A solid white cap in lane 7 was easy to isolate from the background and, despite the low resolution in this lane, the analyst could reliably digitise the swimmer points. This led to the highest intra-analyst precision. Contrast this with the swimmer is lane 6: the red cap merged with the background, this led to it being hard to reliably isolate from the background and therefore digitise. A low precision resulted.

Lane 3, and in particular the segment between 15 m and 35 m in the X axis, had the lowest intra-analyst precision. In this segment, two backstroke swimmer points had standard deviations of 0.81 pixels and 0.56 pixels, which were the highest in the study. These two swimmer points are shown in Figure 10.6. The reason for this low precision was not initially apparent. There was sufficient resolution as it was one of lanes that were closest to the camera, a solid cap colour was used, and the head was not merged with a background object nor completely obscured by splash. Therefore, inconsistency on the part of the analyst whilst digitising these swimmer points was probably the main factor.

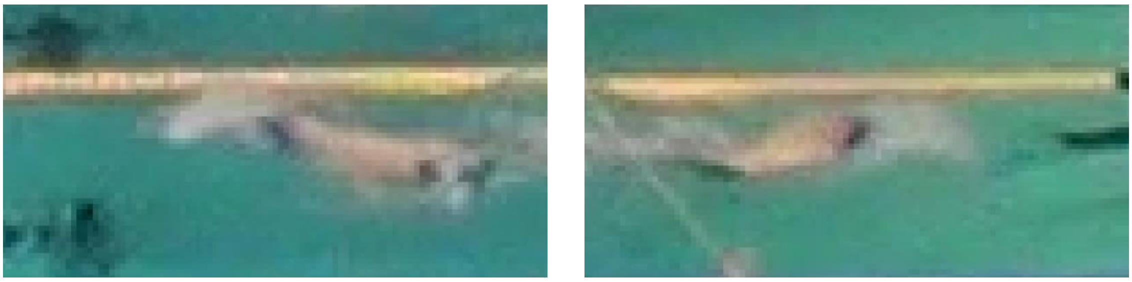


Figure 10.6. Two backstroke swimmer points in the central 20 m segment of lane 3. They had the highest standard deviation in the intra-analyst precision study.

In conclusion, these findings led to the use of a single distribution to describe the uncertainty of all swimmer points in the computer simulation used to calculate swimmer point error. A regular spatial pattern of swimmer point uncertainty was not found. Instead, factors such as cap colour and human inconsistency dominated. As these could not be predetermined, it was assumed that each swimmer point, irrespective of its location on the calibrated plane, would be subjected to the same uncertainty.

10.4.2 Simulation

The variability of the swimmer point error was similar to that found by Veiga et al. (2010). They found a standard deviation of between 15.5 x 10^{-3} m and 34.3 x 10^{-3} m, dependent on lane. This was for a calibrated plane that measured between 10 m and 15 m in the X axis. For the four single-camera viewpoints simulated in this chapter, the standard deviations of the swimmer point errors were between 8.3 x 10^{-3} m and 44.2 x 10^{-3} m. This showed that increasing the size of the calibrated plane in the X axis did not have a large impact on swimmer point error.

Veiga et al. (2010) did not report their intra-analyst precision results. Instead, they just reported swimmer point error. This made it difficult to determine if their swimmer point error was due to intra-analyst precision alone or a combination of intra-analyst precision and resolution, which is an integral part of reconstruction. Chapter 4 showed that for a multi-camera viewpoint similar to that used by Veiga et al. (2010), resolution decreased as distance from the camera increased; this is shown in Figure 4.3 and Figure 4.4. In a low resolution region, a certain sized error in pixels is a larger error in metres than it is in a high resolution region. So, a decrease in resolution would lead to an increase in RDE for the same digitisation error. Therefore, it seemed most likely that the intra-analyst precision found by Veiga et al. (2010) was roughly similar for each lane and that inter-lane differences in reconstructed distances were due to differences in resolution. This was the same as that found by the computer simulation in this chapter.

V1, V2 and V4 were the perpendicular viewpoints. V1 and V2 were fisheye viewpoints and V4 was the non-fisheye one.

There were two general trends in the pattern of swimmer point error. It increased with distance from the camera for all viewpoints and for V1, V2 and V4 it also increased with distance from the optical axis, i.e. 25 m in the X axis. The positive correlation between swimmer point error and distance from the camera could be explained by resolution. However, the pattern of swimmer point error for V1, V2 and V4 did not match that of the resolution. For example, V4’s resolution in the swimming and resultant direction was roughly constant at each distance from the camera (as shown in Figure 4.3 and Figure 4.4). This did not match the pattern of swimmer point error in Figure 10.5. So, a different explanation was sought. The images of the calibrated planes in Figure 3.5 showed that the pixel and world coordinate systems were quite different. In an image, the pixel coordinate system was a uniform, rectilinear and orthogonal grid whereas the world coordinate system was not; these features were lost when it was projected into the image. As shown in Figure 10.7, something similar happened to the pixel coordinate system during reconstruction. A consequence of this was that regular shapes in the pixel coordinate system did not always maintain their shape when reconstructed.

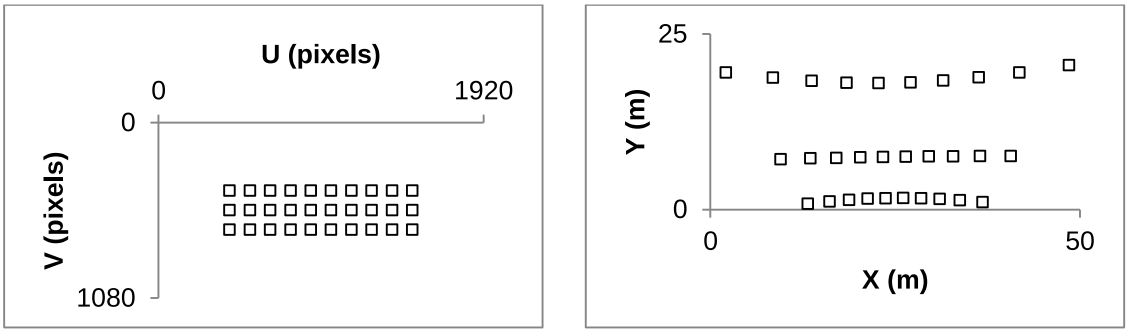


Figure 10.7. Illustration of the effect of reconstruction on a regular grid of points. A calibration model for a camera with a 0.30 multiplier fisheye lens converter was used to do the reconstruction.

In the pixel coordinate system, the uncertainty added to each swimmer point was roughly circular. When reconstructed, it was not. In particular, toward the edges of the calibrated plane the range of the uncertainty in the X axis was larger than it was toward the centre. This is shown in Figure 10.8. So, at the centre of the calibrated plane, the added uncertainty had a smaller effect on reconstructed distances than it did toward the edges. This led to the observed pattern in swimmer point error shown in Figure 10.5.

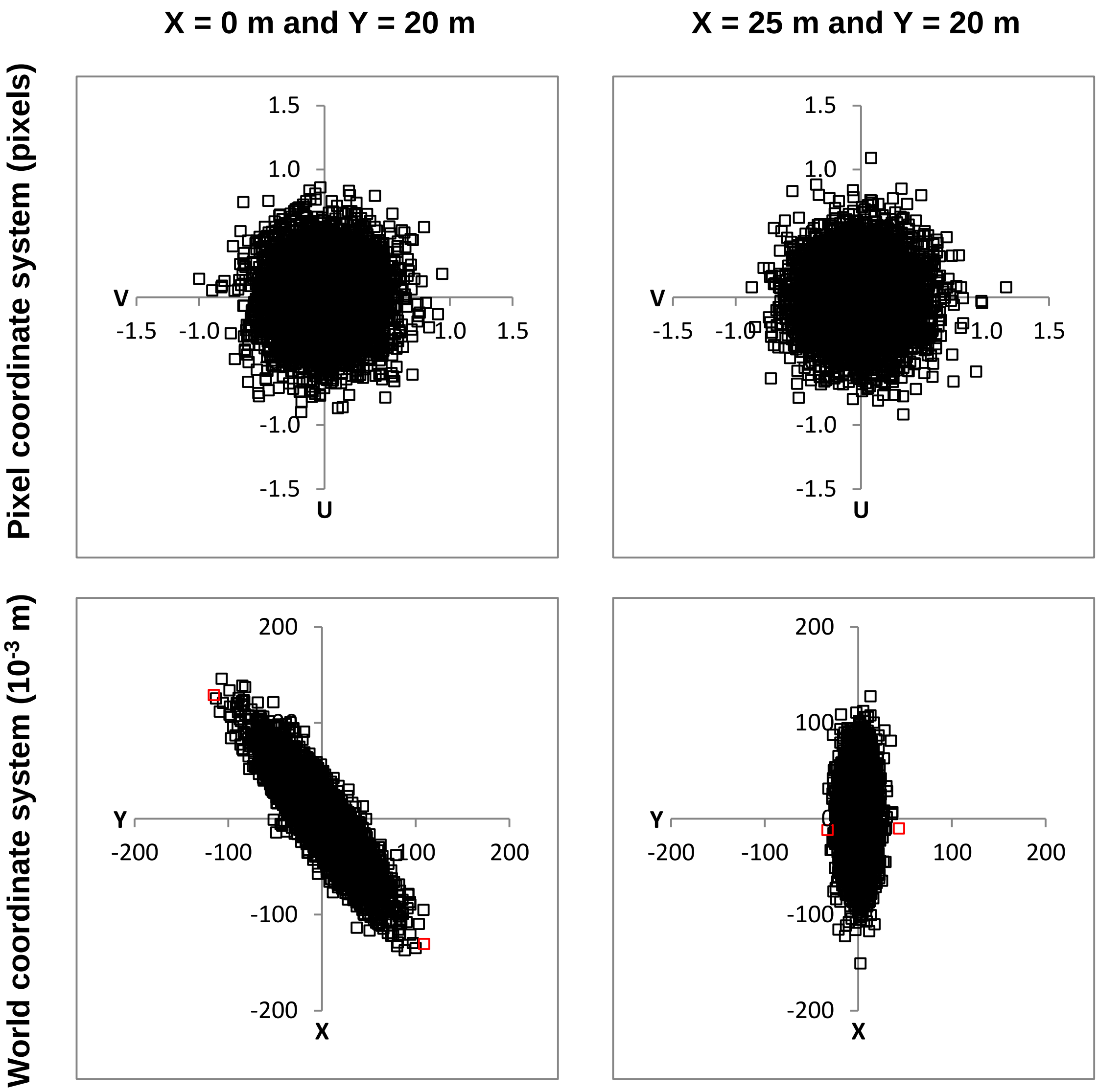


Figure 10.8. Examples of uncertainty added to a swimmer point. The red squares show the range of the X coordinate, which were 225 x 10^{-3} m and 76 x 10^{-3} m.

It is noteworthy that the largest swimmer point errors were, in general, in the region of the calibrated plane occupied by the start, turn and finish phases. As a result, measurement of distances for the clean swimming phase would be less affected by uncertainty in a swimmer point's pixels coordinates than would measurements of the other phases.

V3 was a non-perpendicular and non-fisheye viewpoint.

The largest swimmer point error was for V3. It was -336.4 x 10^{-3} m and occurred in the test distance between 40 m and 45 m in the X axis and at 25 m in the Y axis. The magnitudes of the uncertainties added to the test distance's endpoints were 1.05 pixels and 0.46 pixels. These were 3.75 and 1.64 times the standard deviation found in the intra-analyst precision study. The pattern of absolute swimmer point error for this trial is shown in Figure 10.9. A combination of factors led to this error: the test distance was in a low resolution region of the calibrated plane and the uncertainties were large and both acted to shorten the test distance; the endpoint at 40 m in the X axis was reconstructed as 40.2328 m and the one at 45 m in the X axis as 44.8964 m.

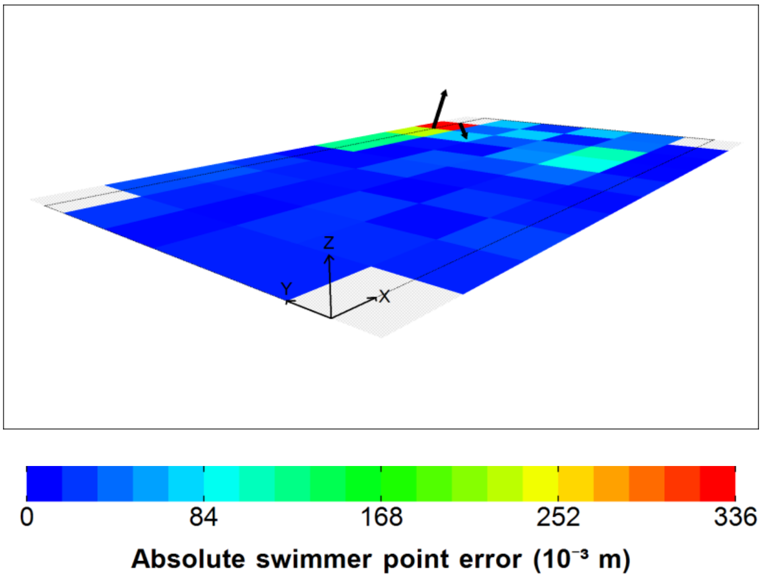


Figure 10.9. Trial that had the highest swimmer point error. Black arrows show the uncertainty; these were scaled (100:1) for illustration purposes. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed. n = 1 for each of the fifty six squares in the heat map.

10.4.3 Limitations

A limitation of this chapter is that it did not assess the effect on swimmer point error of inter-analyst precision. The utilised approach was consistent with that of Veiga et al. (2010).

10.5 Conclusion

This chapter calculated swimmer point error and found:

- Swimmer cap colour and inconsistency on the part of the analyst were major causes of uncertainty in swimmer digitisation.
- The effect this uncertainty had on RDE for a single-camera viewpoint was similar to that reported in the literature for a multi-camera viewpoint.
- Increasing the size of the calibrated plane in the swimming direction did not lead to higher swimmer point error.
- The choice of single-camera viewpoint and the effect this had on reconstruction was the main factor in the pattern and magnitude of the swimmer point error.
- If the swimming competition venue allowed it, the best viewpoint was perpendicular and non-fisheye. This viewpoint had the lowest swimmer point error.
- A perpendicular and fisheye viewpoint should be chosen over a non-perpendicular and non-fisheye one.
- For all viewpoints, swimmer point error was likely to have a smaller impact on the measurements of distance in the clean swimming phase than it would in the other phases.

In the next chapter the swimmer point error quantified in this chapter is combined with the calibration model and control point errors from Chapters 3, 5 and 7.

11 Total error

11.1 Introduction

This chapter calculated total error. This was the combination of all other errors. It showed how errors in the construction of a calibrated plane and digitisation of swimmers on that plane propagated to error in reconstructed distances. Figure 11.1 shows its place in the hierarchy of errors.

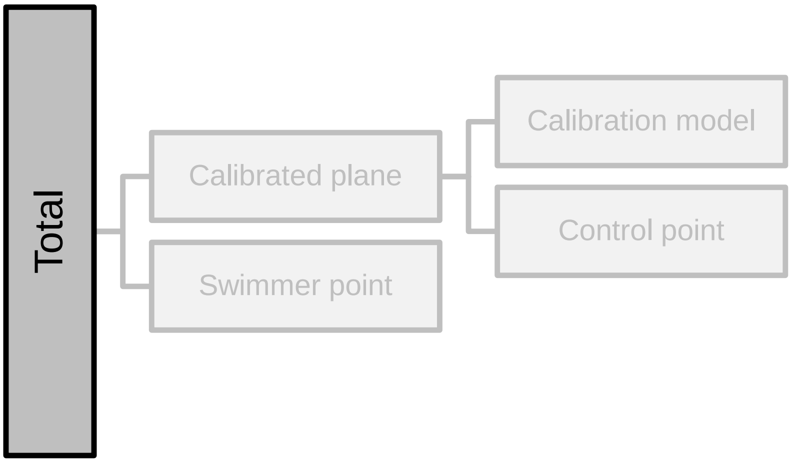


Figure 11.1. Chapter 11 calculated total error.

Total error was calculated using a revised version of the computer simulation first described in Chapter 5. The main change to this simulation was the inclusion of calibration model and swimmer point error. This meant that RDE was a combination of calibrated plane and swimmer point errors.

This chapter also calculated how total error propagated to error in measurement of a swimmer’s speed. Whether a meaningful change in speed could be identified, given the error in the measurement, was of interest. Pyne, Trewin and Hopkins (2004) found that an enhancement in race time of 1% both in the year leading up to a major competition and in the competition itself (i.e. between heat and final) ensured that a swimmer stayed in contention for a medal. So, this chapter assessed whether an enhancement of 1% could be identified given the likely errors in speeds caused by total error. It was assumed that a swimmer would make an equal enhancement in all parts of the race, which would sum to a 1% enhancement in race time.

11.2 Method

The study described in Chapter 5 was repeated, with the following modifications:

- Inclusion of calibration model error. When calculating control point error in Chapters 5 and 7 the test distances had projected pixel coordinates. This removed calibration model error. In this chapter, the test distances had the pixel coordinates that were used when calculating calibration model error in Chapter 3.
- Control point pixel coordinate system uncertainty. In Chapter 5, one distribution was used. In this chapter, one distribution per viewpoint was used. Each was a Gaussian normal distribution with a mean of zero pixels and a standard deviation given by the results from Chapter 6, i.e. 0.11 pixels, 0.11 pixels, 0.14 pixels and 0.07 pixels for V1, V2, V3 and V4 (respectively).
- Inclusion of swimmer point error. Uncertainty in a swimmer point's pixel coordinates were modelled by the distribution from Chapter 10, i.e. a Gaussian normal with a mean of zero pixels and a standard deviation of 0.28 pixels.
- Uncertainty was added to the pixel and world coordinates of the control points and the pixel coordinates of the swimmer points.

- Test distances of different lengths. In previous chapters test distances of 5 m were used. In this chapter test distances of length 5 m to 50 m in 5 m increments were used.
- RDE was called total error in this chapter.
- Calculation of speed. Speed was calculated using two methods: two-point and simple linear regression. The two-point method calculated speed as *d' / t*, where *d'* was the reconstructed distance and *t* the time to swim the ground truth distance at the ground truth speed. The simple linear regression method used the reconstructed X coordinates of all swimmer points on a track between and including a test distance's endpoints; the gradient of the estimated line gave the swimmer speed.
- Calculation of error in speeds. Speed error for a test distance was the difference between its reconstructed and ground truth speeds. Speed error was calculated for a baseline speed of 1.6 m s^{-1} and an enhanced speed of 1.616 m s^{-1}, i.e. a 1% enhancement of the baseline speed. The baseline speed was the mean of those calculated in Chapter 9 (when rounded to one decimal place).
- Assessment of whether a 1% enhancement in speed could be identified. The method described by Bland and Altman (1984) was used to calculate a lower limit, upper limit and limits range for baseline and enhanced speed for each test distance length in each viewpoint. If there was a gap between the limits range for baseline and enhanced speeds, then an enhancement in swimmer speed could be identified. This is illustrated in Figure 11.2.

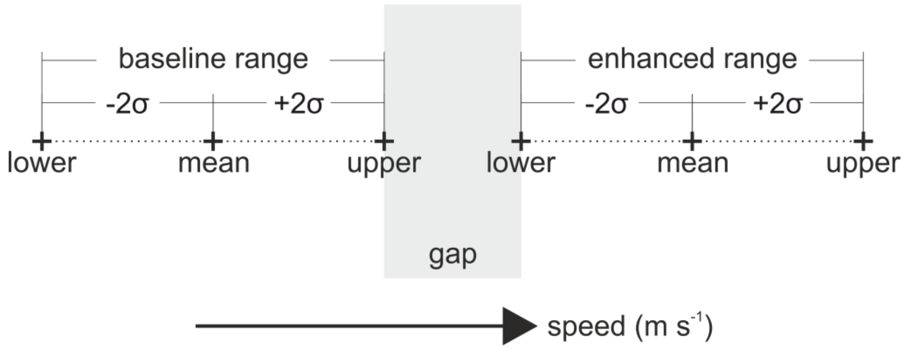


Figure 11.2. Identifying a 1% improvement in speed. If there was a gap between the limits range for baseline and enhanced speeds then an enhancement could be identified. If the ranges overlapped, then speeds in the overlap could be due to either speed error or an enhancement.

11.3 Results

11.3.1 Total error

The total error for all viewpoints and lengths of test distance are in Appendix 15.3. For test distances of length 5 m, the mean ± standard deviation of the total error and the errors that were combined to make it are in Table 11.1. The mean of the total error and the calibrated plane error were the same, but their standard deviations differed: total error's was larger.

Table 11.1. Mean ± standard deviation of total error and the errors that were combined to make it. These results are for test distances of length 5 m.

	n	V1	V2	V3	V4
Calibrated plane (10^{-3} m)	560000	10.3 ± 12.8	6.0 ± 19.7	-6.0 ± 38.8	0.3 ± 4.6
Swimmer point (10^{-3} m)	560000	0.0 ± 17.4	0.0 ± 16.9	0.0 ± 40.5	0.0 ± 9.6
Total (10^{-3} m)	560000	10.3 ± 21.6	6.0 ± 26.0	-6.0 ± 56.2	0.3 ± 10.6

The pattern of the mean of the absolute total error in Figure 11.3 was similar to that for calibrated plane error shown in Figure 8.2.

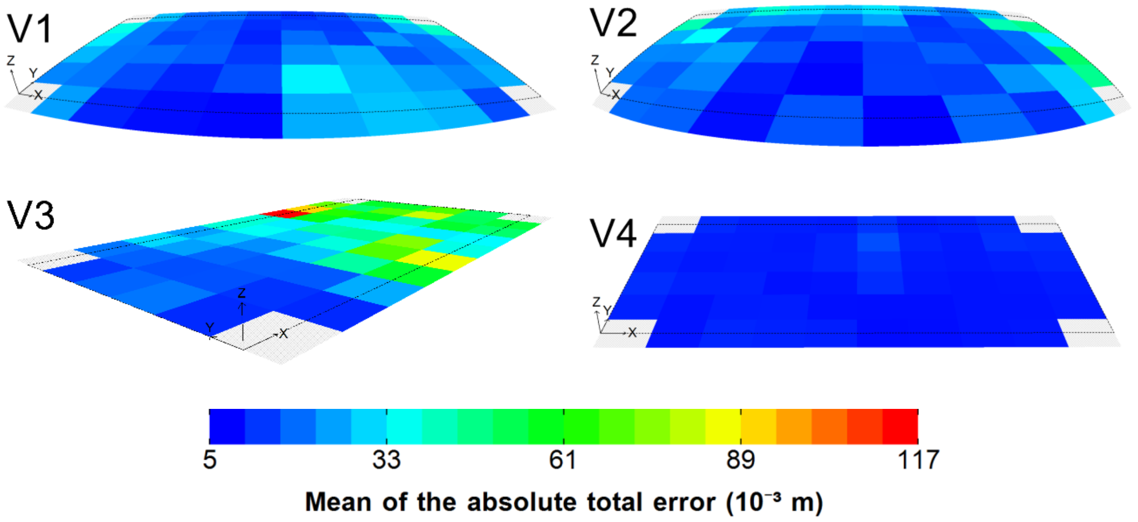


Figure 11.3. Pattern of the mean of the absolute total error for test distances of length 5 m. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed. n = 10000 for each of the 56 squares in each heat map.

The pattern of the standard deviation of the total error in Figure 11.4 was similar to that of the swimmer point error shown in Figure 10.5.

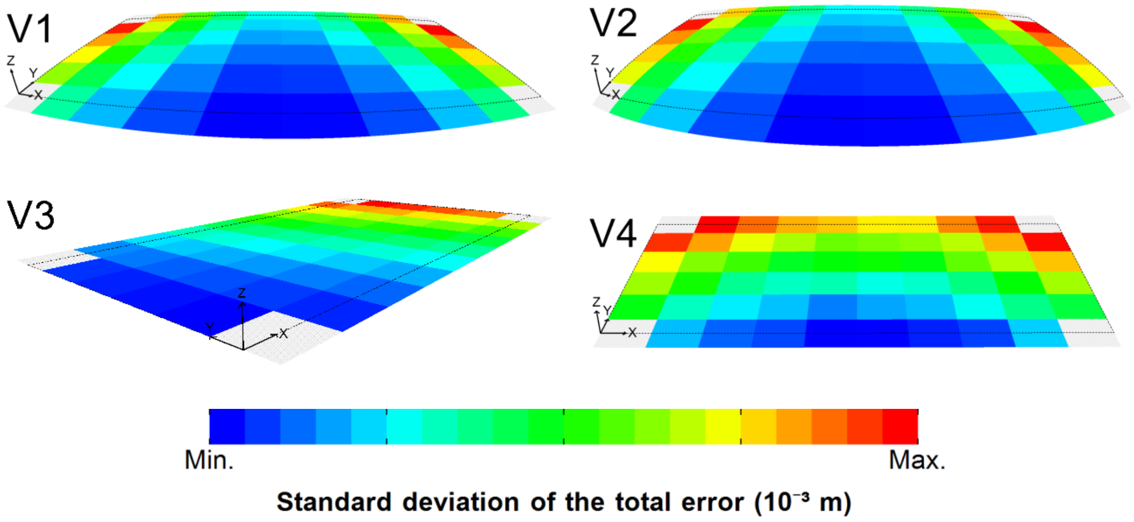


Figure 11.4. Pattern of the standard deviation of the total error. The black dotted line is the outline of the calibrated plane. The distances in the grey hatched areas at the corners were not reconstructed. n = 10000 for each of the 56 squares in each heat map. Each heat map had its own scale with blue representing the viewpoint's minimum error and red the maximum.

11.3.2 Speed error

The speed error for all viewpoints and lengths of test distance are in Appendix 15.4 (for the two-point method) and Appendix 15.5 (for the simple linear regression method).

The mean ± standard deviation of the absolute differences between the speeds calculated by the two methods is shown in Table 11.2. The differences were small.

Table 11.2. Mean ± standard deviation of the absolute difference between speeds calculated by the two-point and simple linear regression methods.

Test distance length (m)	n	Difference between methods (m s^{-1})			
		V1	V2	V3	V4
5	560000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
10	500000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
15	440000	0.000 ± 0.001	0.000 ± 0.001	0.001 ± 0.001	0.000 ± 0.000
20	380000	0.000 ± 0.001	0.001 ± 0.001	0.001 ± 0.002	0.000 ± 0.000
25	320000	0.001 ± 0.001	0.001 ± 0.001	0.001 ± 0.002	0.000 ± 0.000
30	260000	0.001 ± 0.001	0.001 ± 0.001	0.002 ± 0.002	0.000 ± 0.000
35	200000	0.001 ± 0.001	0.001 ± 0.001	0.002 ± 0.002	0.000 ± 0.000
40	140000	0.001 ± 0.001	0.001 ± 0.001	0.002 ± 0.002	0.000 ± 0.000
45	80000	0.001 ± 0.001	0.001 ± 0.001	0.002 ± 0.002	0.000 ± 0.000
50	40000	0.001 ± 0.001	0.001 ± 0.001	0.001 ± 0.002	0.000 ± 0.000

11.3.3 Identifying a 1% enhancement in swimmer speed

The 95% limits of agreement for all viewpoints and lengths of test distance are in Appendix 15.6 (for the two-point method) and Appendix 15.7 (for the simple linear regression method).

For a 1% enhancement to be identified required the standard deviation (σ) of the speed error to be less than 0.004 m s^{-1}. This gave a 2σ (i.e. a 95% limit of agreement) of 0.008 m s^{-1}, which was half the gap between baseline and enhanced speeds. The bias did not affect this as it was the same for baseline and enhanced speed for each combination of viewpoint and length of test distance. A visualisation of the gaps for all test distance lengths and both methods of calculating speed is in Table 11.3.

Table 11.3. Gap, in m s^{-1}, between baseline and enhanced speeds for test distances of length 5 m to 50 m. Leftmost column for each viewpoint is the gap for the two-point method and rightmost is the gap for the simple linear regression method.

	V1		V2		V3		V4	
5	-0.012	-0.012	-0.018	-0.017	-0.056	-0.056	0.002	0.002
10	0.001	0.001	-0.005	-0.004	-0.036	-0.035	0.008	0.008
15	0.005	0.005	0.000	0.001	-0.029	-0.029	0.010	0.010
20	0.007	0.006	0.002	0.004	-0.024	-0.025	0.011	0.011
25	0.007	0.008	0.004	0.005	-0.019	-0.021	0.012	0.012
30	0.009	0.009	0.006	0.006	-0.016	-0.018	0.012	0.012
35	0.009	0.009	0.006	0.007	-0.012	-0.015	0.013	0.013
40	0.009	0.010	0.007	0.009	-0.007	-0.012	0.013	0.013
45	0.009	0.010	0.009	0.011	-0.001	-0.006	0.014	0.014
50	0.011	0.011	0.009	0.011	-0.001	-0.004	0.014	0.014

A 1% enhancement in speed could not be identified in V3. There was one key difference between the two methods of calculating speed: for V2, simple linear regression could identify a 1% enhancement over 15 m or more whereas two-point required 20 m or more.

11.4 Discussion

11.4.1 Total error

Total error represented the RDE that was likely to occur during competition performance analysis. It combined errors in the calibration model, the control points, and the digitisation of the swimmers. How each of these constituent errors contributed to the bias and variability in RDE is visualised in Figure 11.5.

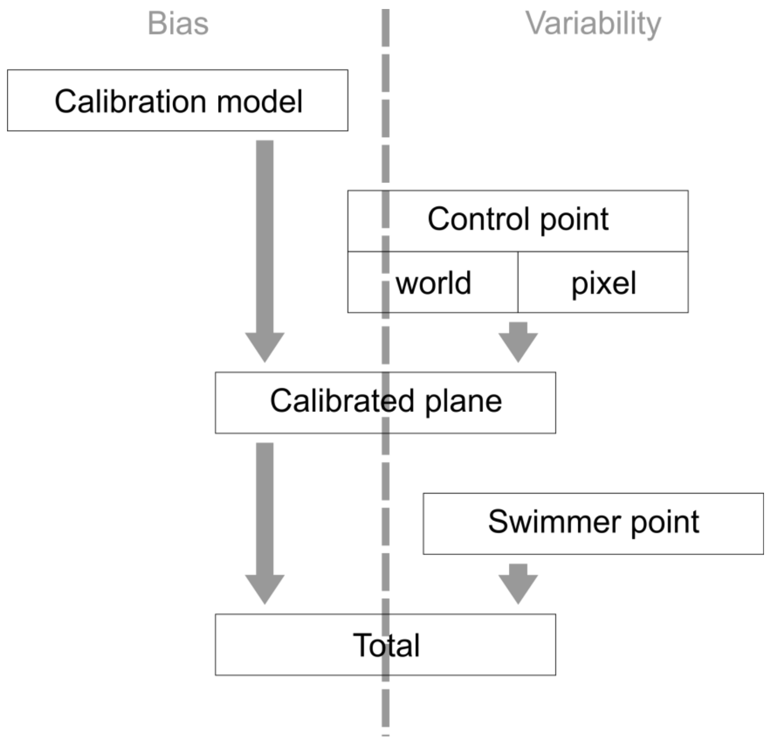


Figure 11.5. Constituent errors contribution to total error's bias and variability.

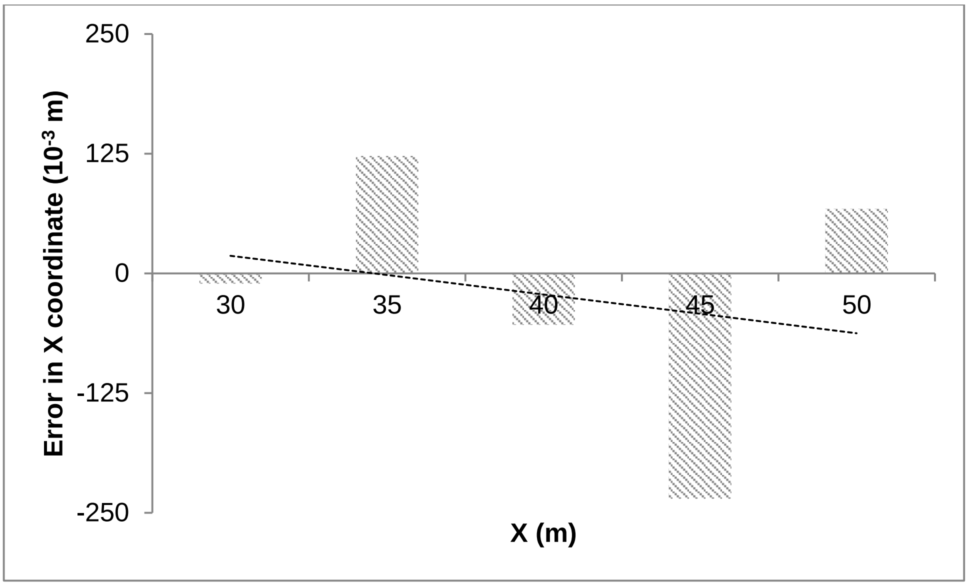
The main source of bias was calibration model error and the main source of variability was swimmer point error. Figure 11.3 and Figure 11.4 confirmed this: the mean of the absolute total error had the same pattern as the calibrated plane error in Figure 3.7 and the calibration model error in Figure 8.2. The standard deviation of the total error had the same pattern as the swimmer point error in Figure 10.5. As a consequence, control point error contributed relatively little to total error.

11.4.2 Speed error

There was little difference between the two methods of calculating speed. This showed that simple linear regression did not minimise the effect on reconstructed speed of error in the calibrated plane and the swimmer points. One likely factor in this was the number of coordinates that simple linear regression had to work with. For test distances of length 5 m there were two and this made two-point and simple linear regression identical. At maximum, there were eleven coordinates. This number of points may have provided insufficient spatial information to minimise the effect of total error on calculated speeds.

V3 was the non-perpendicular and non-fisheye viewpoint.

For the baseline speed, the largest difference between two-point and simple linear regression methods was for V3. It was for the test distance at 15 m in the Y axis and between 30 m and 50 m in the X axis. The difference between the two methods was 0.012 m s^{-1}: two-point calculated 1.606 m s^{-1} and simple linear regression 1.594 m s^{-1}. The errors in the reconstructed X coordinates used to calculate these speeds are shown in Figure 11.6. The X coordinate at 30 m had a small negative error and the one at 50 m had a positive error. This led to an overestimation of speed by the two-point method as the reconstructed distance was approximately 20.08 m. The largest errors in the reconstructed X coordinates were at 35 m and 45 m. Consequently, they did not affect the two-point method, but they did have a large influence on the simple linear regression method. The error at 35 m pulled the line of best fit upward whilst the one at 45 m pushed it downward. As a result, the simple linear regression method underestimated the ground truth speed.



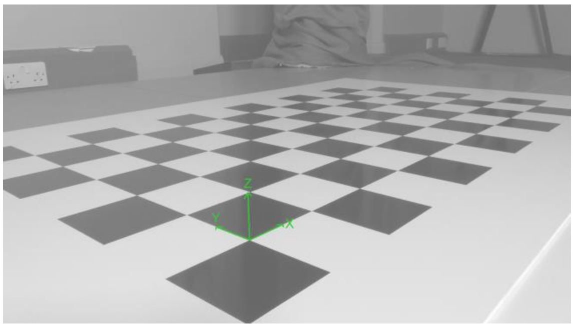


Figure 11.6. Error in the reconstructed X coordinate of the swimmer points at 15 m in the Y axis and between 30 m and 50 m in the X axis. The data is for the trial that led to the largest difference between the two methods of calculating speed, which was for V3. The black dashed line shows the simple linear regression.

This example highlighted how the simple linear regression method could calculate accurate and precise speeds from noisy coordinates. The simple linear regression method was subjected to the two largest errors in the reconstructed X coordinates. Notably, these errors were an order of magnitude higher than one of the two coordinates used by the two-point method. Nevertheless, the effect of these was minimised to give the same absolute speed error as the two-point method.

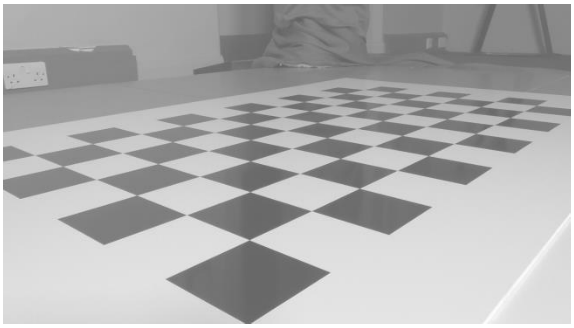
11.4.3 Identifying a 1% enhancement in swimmer speed

V3 was the non-perpendicular and non-fisheye viewpoint.

A 1% enhancement in speed could not be identified in V3. However, if 90% limits of agreement are considered instead of 95% ones then it could be identified for test distances of length 45 m or more. This is shown in Table 11.4.

Table 11.4. 90% limits of agreement for V3. The two-point method was used to calculate speed.

Test distance length (m)	Baseline (m s^{-1})	Enhanced (m s^{-1})	Gap (m s^{-1})
35	1.587 to 1.611	1.603 to 1.627	-0.008
40	1.589 to 1.608	1.605 to 1.624	-0.004
45	1.591 to 1.606	1.607 to 1.622	0.001
50	1.592 to 1.606	1.608 to 1.622	0.002



11.5 Conclusion

This chapter calculated total error and found:

- Total error was found to be a straightforward combination of its constituent errors: calibrated plane and swimmer point. Calibrated plane error determined the bias and swimmer point the variability.
- Control point error had a relatively minor effect on total error.
- All the perpendicular viewpoints had lower variability speed error than the non-perpendicular one. As a result, a 1% enhancement in speed could be identified in the perpendicular viewpoints but not in the non-perpendicular one.

- If the swimming competition venue allowed it, the best viewpoint was perpendicular and non-fisheye. This viewpoint had the lowest total and speed errors and could identify a 1% enhancement in speed for distances of 5 m or more.
- A perpendicular and fisheye viewpoint should be chosen over a non-perpendicular and non-fisheye one.

In the next chapter, a method of calculating clean swimming speed using automated digitisation is developed.

12 Clean swimming speed from automated digitisation

12.1 Introduction

This chapter developed an automated digitisation system and compared its results with those from the manual methods used in Chapter 9. The starting point for this system was iSwim (Driscoll and Kelley 2013). This represented the closest match to the desired system, i.e. it automatically digitised swimmers in the clean swimming phase of a race from video footage captured from a single-camera viewpoint. iSwim’s background model was a single image when the pool was empty. Differencing and thresholding of the RGB_R values on five profile lines, which ran down the centre of each lane, was used to identify the swimmers. A weakness in iSwim was that it did not account for the highly dynamic nature of a swimming pool: movements such as ripples, splashes and reflections were not modelled. The system developed in this chapter did; a technique from the drowning prevention systems (such as Eng et al. 2003) was used.

12.2 Method

12.2.1 Video footage

The video footage and PNG files used in Chapter 9 were also used in this chapter. The footage was split into a training set and a test set. The training set was used to develop the automated digitisation system and the test set was used to assess it. The men’s 400 m individual medley was the training set and the other eight races were the test set.

12.2.2 Regions of interest

Nine regions of interest (ROI) were defined. The first ROI contained the central eight swimming lanes and the other eight were for the individual lanes. Projection was used to calculate an ROI's pixel coordinates. The eight lane ROI and the ROI for lane one are shown in Figure 12.1.

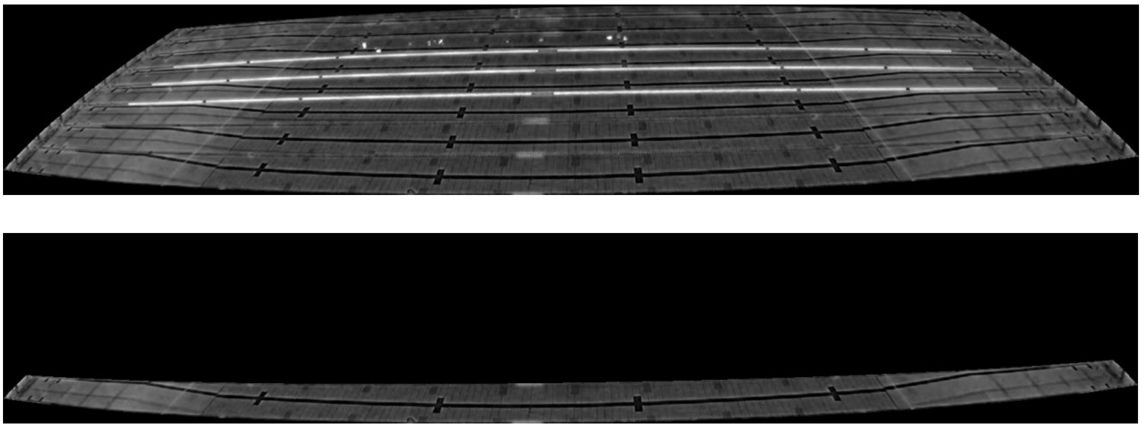


Figure 12.1. Regions of interest (ROI) for the central eight lanes and one of the eight lanes.

12.2.3 Digital image processing

The automated digitisation system was implemented as a Microsoft C#.NET (Microsoft 2012) application. For each frame f_i, the following steps were performed:

1. Read the pixels in the eight-lane ROI for frame f_{i-k}, f_i and f_{i+k} ($k = 100$) into memory. Only the RGB$_r$ values were stored in memory.
2. Created a background image, B_i, using a temporal median filter. An example of how this worked is shown in Figure 12.2.

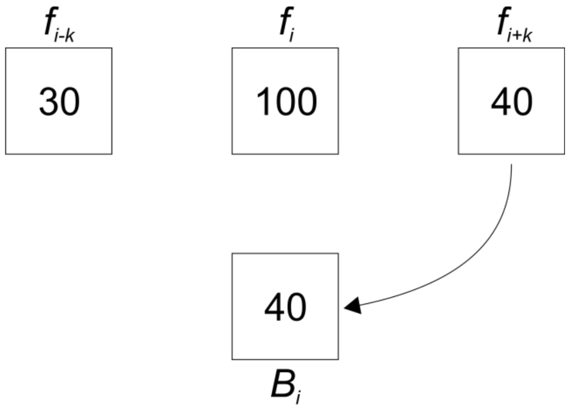


Figure 12.2. Operation of a temporal median filter for one pixel in frames f_{i-k}, f_i and f_{i+k}.

The pixel in frames f_{i-k}, f_i and f_{i+k} had RGB_r values of 30, 100 and 40. The temporal median filter found which of these values had the smallest sum of differences from the other two using Equation 2.25. The results of the filter's calculations are shown in Table 12.1. As f_{i+k} had the smallest sum, the background image pixel had value 40.

Table 12.1. Sum of differences between RGB_r values for frames f_{i-k}, f_i and f_{i+k} in Figure 12.2.

	f_{i-k}	f_i	f_{i+k}	Sum
RGB_r	30	100	40	-
Diff. f_{i-k}	0	70	10	80
Diff. f_i	70	0	60	130
Diff. f_{i+k}	10	60	0	70

The construction of a background image using a temporal medial filter is illustrated in Figure 12.3. The three frames that were used to construct the background image, i.e. f_{i-k}, f_i and f_{i+k}, are shown in this figure.

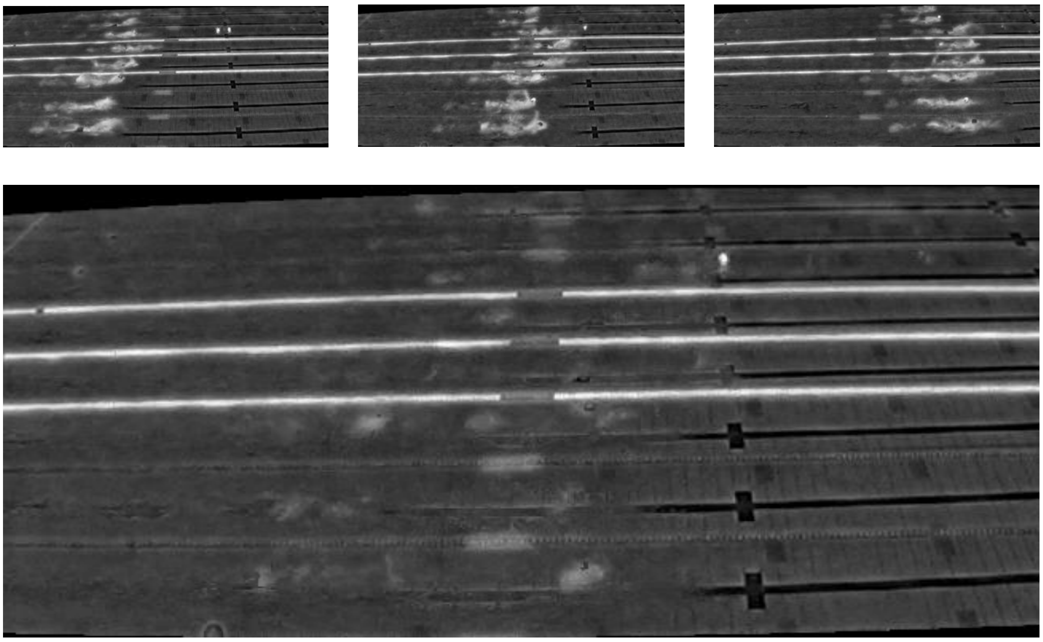


Figure 12.3. An extract from a background image that was constructed from three frames using a temporal median filter.

3. Subtracted frame f_i from B_i. This was a pixel-by-pixel subtraction, e.g. pixel (0, 0) in f_i was subtracted from pixel (0, 0) in B_i. If a result was negative then that pixel's intensity was set to zero. In this way, only those pixels whose RGB_r was greater in f_i than it was in B_i were greater than zero in the resulting difference image. An example difference image is in Figure 12.4.

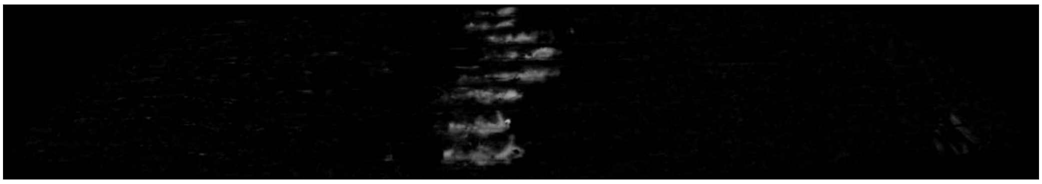


Figure 12.4. Difference image.

4. Thresholded the difference image at an intensity level selected by Otsu's automated method (Otsu 1979). AForge's implementation of this method was used (AForge 2015). The result was an image in which the

background was black and foreground objects, including the swimmers, were white. An example thresholded image is in Figure 12.5.

Figure 12.5. Thresholded image.

5. For each lane, the individual foreground objects in the thresholded image were identified and labelled. AForge's connected region function and the lane ROIs were used to do this. This gave a set of foreground objects per lane. Figure 12.6 shows two of these for lane one. Each object had information such as location and size (e.g. area and bounding rectangle). The foreground object with the largest area in pixels was selected and the pixel coordinates of its centroid were reconstructed. This gave the swimmer's world coordinates in each lane in f_i.

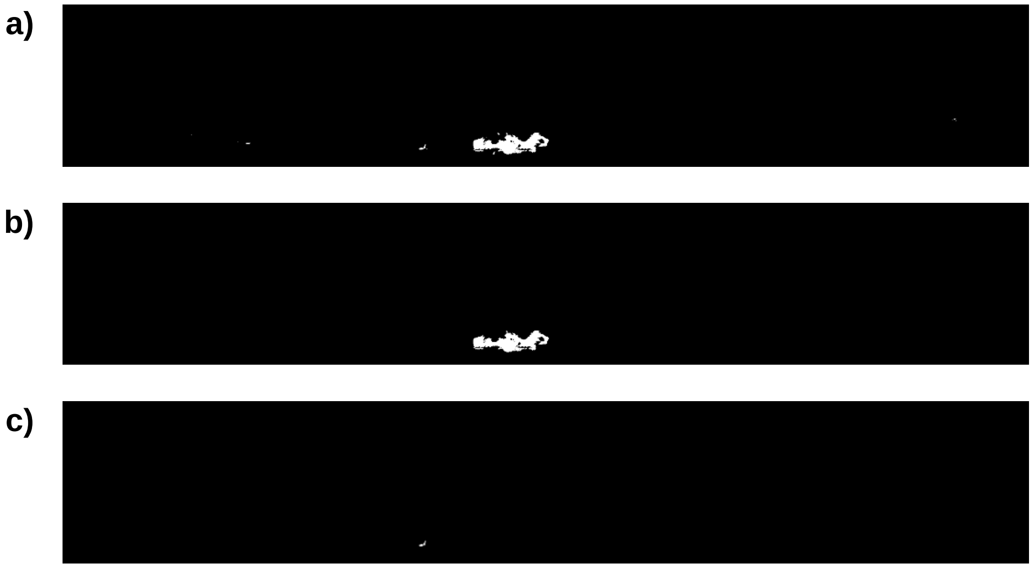


Figure 12.6. Foreground objects in lane 1: a) all foreground objects; b) and c) two of the individual foreground objects.

12.2.4 Calculation of clean swimming speed

The result of steps 1 to 5 was the swimmer's world coordinates in each lane in f_i. These coordinates were used to calculate the swimmer's speed in the clean swimming phase of each lap. The first and last digitised frame for the corresponding swimmer and lap from Chapter 9 was used to split the frames into laps. In this way, the speeds in the manual and automated digitisation studies were calculated for the same time period. The automated digitisation method used all frames in the clean swimming phase whereas manual used only the frames that could be digitised. Speed was calculated in the same way as in Chapter 9. Simple linear regression was used, i.e. a straight line was fitted through the reconstructed X coordinates and the gradient of this line gave the speed. Agreement between the speeds calculated from the manual and automated tracking was assessed using Bland and Altman's (1986) method.

12.3 Results

Clean swimming speeds for all laps in all races are in Appendix 15.8.

The mean ± standard deviation of the execution time for the five digital image processing steps is shown in Table 12.2; this data was for the men's 400 m individual medley race. The automated digitisation system was run on a Shuttle XPC SH67H3 computer, which had an Intel Core i7-2600 3.4 GHz CPU and 4 GB of RAM. The largest execution times were for reading pixels from the PNG files and construction of the background image.

Table 12.2. Mean ± standard deviation of the execution time for the five digital image processing steps.

Operation	Execution time (s)
1. Read pixels	0.216 ± 0.006
2. Background image	0.136 ± 0.003
3. Difference image	0.003 ± 0.001
4. Thresholded image	0.032 ± 0.011
5. Foreground object	0.024 ± 0.010

The mean ± standard deviation of the time to digitise a lap for manual and automated digitisation methods is shown in Table 12.3; this data is for the men’s 400 m individual medley race. The manual method data was from Chapter 9.

Table 12.3. Mean ± standard deviation of the time to digitise a lap for manual and automated digitisation methods.

Stroke	n	Manual time (min:sec)	Automated time (min:sec)
Backstroke	16	14:37 ± 3:55	8:08 ± 0:45
Breaststroke	16	4:29 ± 1:16	9:34 ± 0:47
Butterfly	16	4:35 ± 0:55	7:49 ± 0:55
Freestyle	15	11:41 ± 3:9	8:23 ± 0:41

The agreement between the speeds from the manual and automated digitisation studies for the training and test sets are summarised in Table 12.4.

Table 12.4. Agreement between speeds from the manual and automated digitisation methods.

Set	n	Mean (m s^{-1})	σ (m s^{-1})	Mean ± 2σ (m s^{-1})
Training	63	0.003	0.008	-0.012 to 0.018
Test	141	0.005	0.015	-0.024 to 0.034

In the test set some races had noticeably better agreement between the two methods than did others. This is shown in Table 12.5. The agreement was good for the men’s 100 m breaststroke but not for the women’s 50 m backstroke.

Table 12.5. Agreement between speeds from the manual and automated digitisation methods for the test set.

Race	n	Mean (m s^{-1})	σ (m s^{-1})	Mean ± 2σ (m s^{-1})
Men’s 100 m breaststroke	16	0.001	0.004	-0.008 to 0.009
Men’s 100 m butterfly	16	0.011	0.018	-0.025 to 0.047
Men’s 100 m freestyle	16	0.009	0.016	-0.023 to 0.040
Men’s 200 m backstroke	32	0.004	0.014	-0.023 to 0.032
Women’s 100 m freestyle	13	0.008	0.009	-0.009 to 0.026
Women’s 200 m breaststroke	32	-0.003	0.010	-0.023 to 0.016
Women’s 50 m backstroke	8	0.022	0.025	-0.029 to 0.072
Women’s 50 m butterfly	8	0.008	0.011	-0.014 to 0.030

The speed profiles produced by the manual and automated digitisation methods were compared using the case studies and plots described in Section 9.4.3. These are shown in Figure 12.7 to Figure 12.9. The plots for the automated method showed higher variability about their regression lines than did the manual one. This was especially apparent for the rate of change of the reconstructed X coordinate (i.e. δX) and residual against distance plots.

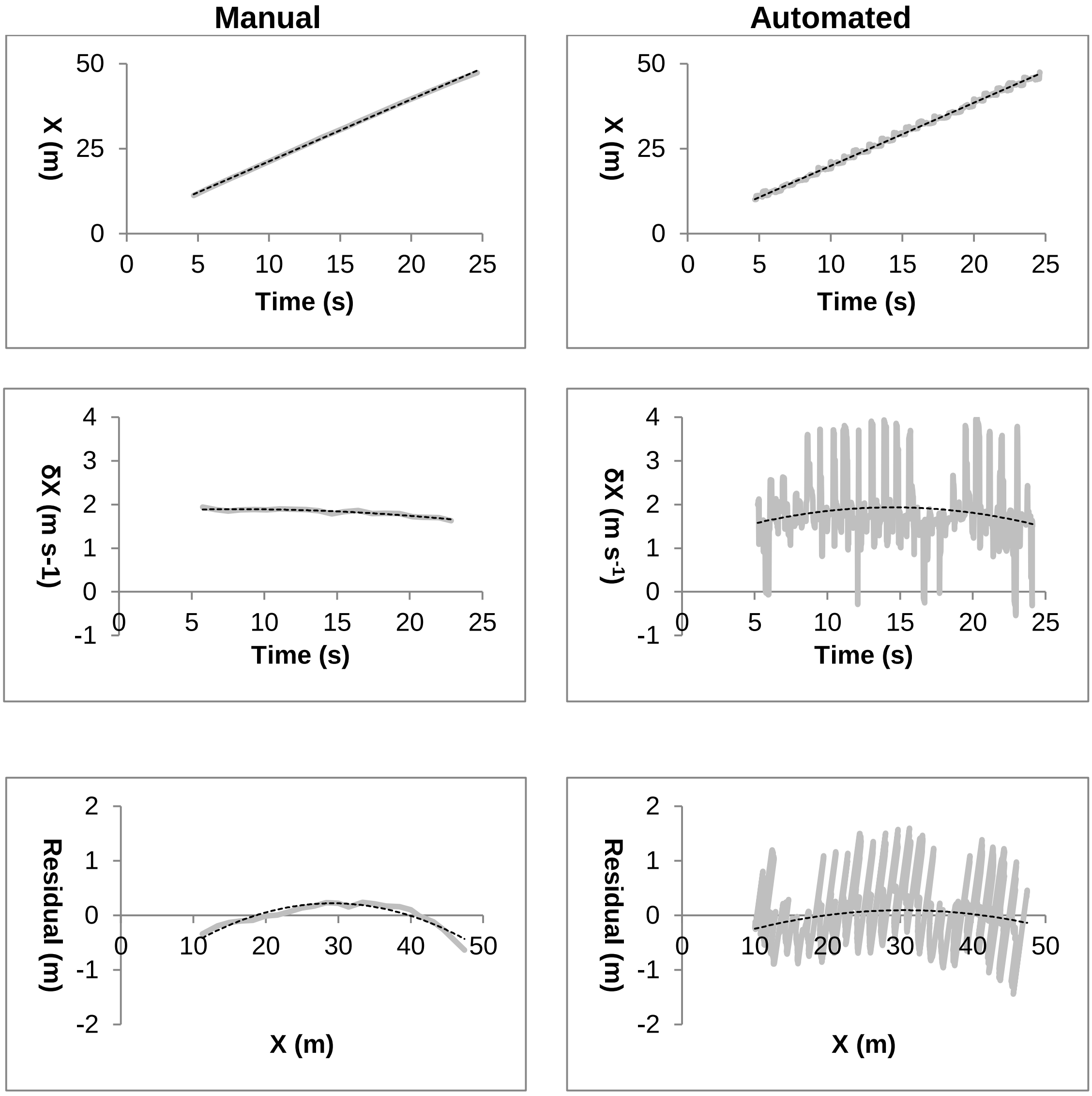


Figure 12.7. Comparison of speed profiles produced by the manual and automated digitisation methods for the women 50 m butterfly case study.

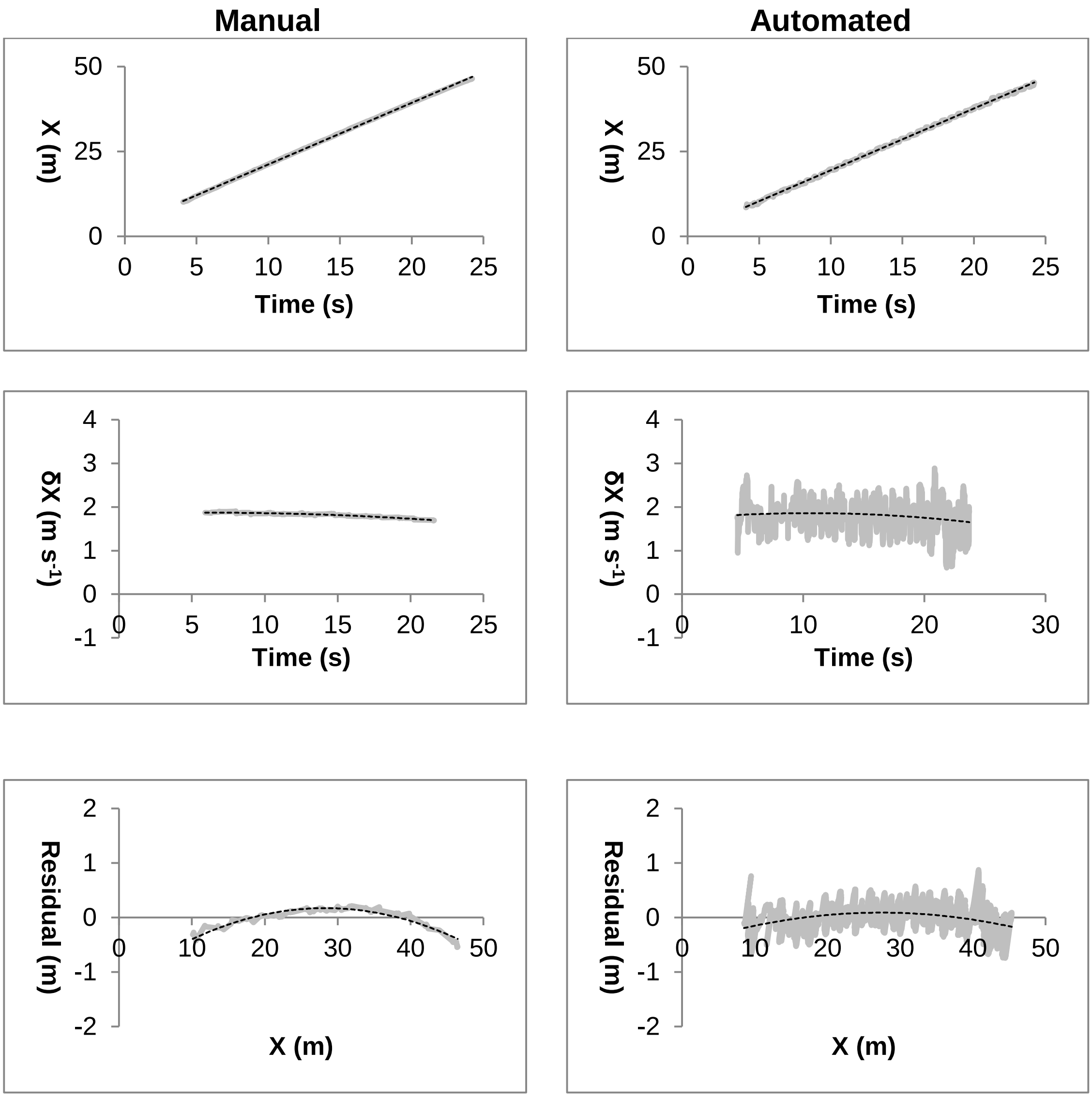


Figure 12.8. Comparison of speed profiles produced by the manual and automated digitisation methods for lap one of the women 100 m freestyle case study.

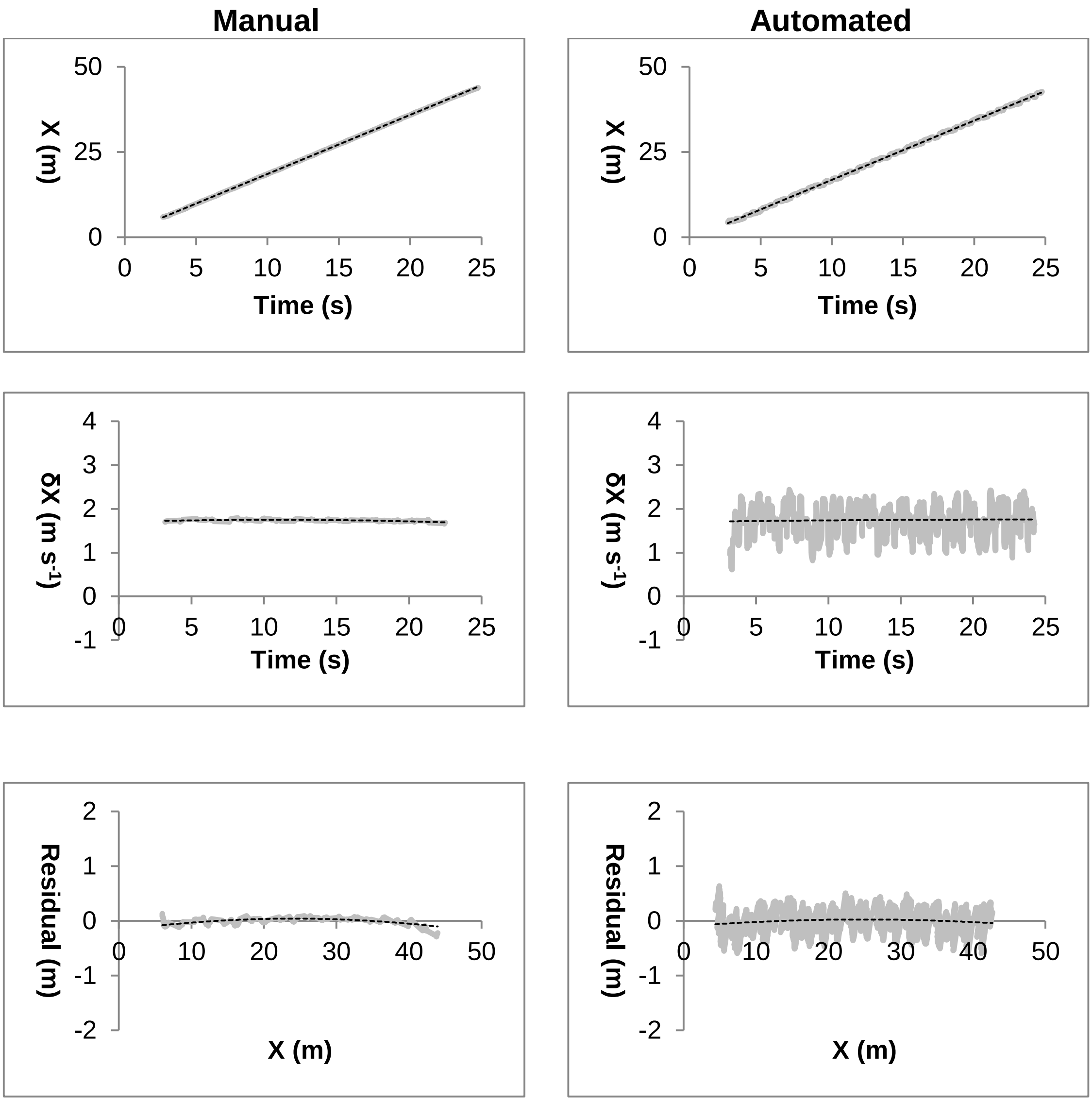


Figure 12.9. Comparison of speed profiles produced by the manual and automated digitisation methods for lap two of the women 100 m freestyle case study.

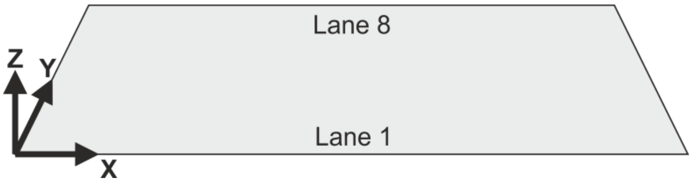
12.4 Discussion

12.4.1 Region of interest

iSwim processed only one lane whereas the automated digitisation system simultaneously processed the swimmers in all lanes. This came at the cost of a higher computation burden: more pixels had to be stored in memory and processed. The ROI for each lane contained all pixels whereas iSwim used five profile lines per lane. Using profile lines could have reduced the computational burden: Table 12.6 shows that over 90% fewer pixels would have been stored and processed if they had been used. However, an ROI that uses all pixels is more traditional, e.g. all the automated swimmer digitisation systems reviewed in Chapter 2, except iSwim, used this approach. In addition, development and testing of a system that used the profile line approach proved to be problematic. Critically, it was difficult to visualise the results of the five digital image processing steps.

Table 12.6. Number of pixels for an ROI that contained all pixels and for five profile lines. Also given is the number of profile line pixels expressed as a percentage of all pixels.

Lane	All pixels	Five profile lines
1	162432	8283 (5.1%)
2	115162	7927 (6.9%)
3	88407	7572 (8.6%)
4	84303	7222 (8.6%)
5	81838	6889 (8.4%)
6	76665	6567 (8.6%)
7	73131	6263 (8.6%)
8	67375	5981 (8.9%)



12.4.2 Background image

The main difference between iSwim and the system developed in this chapter was the model of the background objects. iSwim used a single background image whereas the system developed in this chapter constructed a clean background image for each frame. iSwim's approach proved to be a poor representation of the background objects, especially as time into the race increased. As shown in Figure 12.10, this made it difficult to distinguish the swimmers from the lane ropes and the pool floor markings.

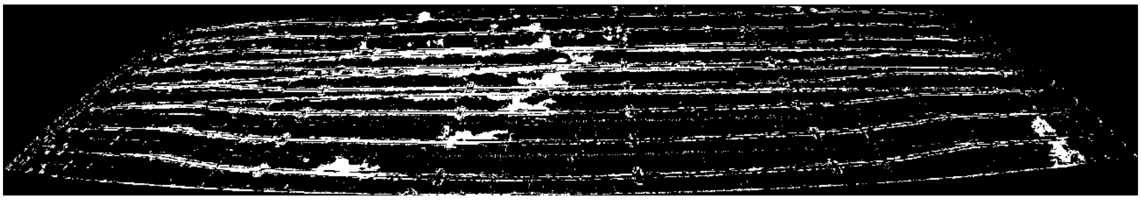


Figure 12.10. Thresholded image that resulted from modelling the background objects with a single static background image.

These background objects had to be removed at some point. A clean background image constructed using a temporal median filter allowed this to be done at an early stage. Eng et al. (2003) established the use of a temporal median filter to construct a clean background image of a swimming pool. They constructed an initial clean background image and then periodically updated it using a weighted averaging scheme. A different approach was used in this chapter: each f_i had its own clean background image, B_i, which was constructed from f_{i-k}, f_i and f_{i+k} (k = 100). The inspiration for this approach was that the best background image would be f_i, but without the swimmers in it. In this way, the image artefacts caused by the unique background motion in f_i would be captured. This intuition led to the adopted approach: the temporal median filter was centred on f_i and used the smallest k that removed the swimmer from the resultant background image. It was thought that a small k would give f_{i-k} and f_{i+k} that were similar to f_i and hence the desired clean background image, i.e. f_i but without the swimmers. A k of 100 was settled on; this performed adequately for men and women, all strokes and speeds in the range 1.1 m s^{-1} to 2.1 m s^{-1}. A downside of the adopted approach was a high computational burden. Each B_i required three images to be read from hard disk and stored in memory. In addition, the temporal median filter required the processing of each pixel in the eight-lane ROI. The burden could have been reduced by using a B_i for more than one f_i. Further testing would be required to ascertain how this would affect the subsequent processing steps.

12.4.3 Difference image

iSwim classified any change in RGB_r as a potential swimmer whereas the method presented in this chapter looked only for an increase in RGB_r. This was consistent with the iSwim results, i.e. the presence of a swimmer is characterised by an increase in RGB_r (Driscoll and Kelley 2013, p16).

12.4.4 Thresholded image

The aim of thresholding was to classify the objects in the difference image as either foreground or background. It was expected that the swimmer would be classified as a foreground object and everything else would be a background object but this was not the always the case. An example of this is shown in Figure 12.11. It was noted that the non-swimmer objects tended to be smaller than the swimmer one and this led to a simple solution for identifying the swimmer.

Figure 12.11. Potential foreground objects. The largest object is in the yellow rectangle and the other objects are in the red rectangles.

12.4.5 Identifying swimmers

In each lane, the object in the thresholded image that had the largest area in pixels was assumed to be the swimmer. In general, this simple approach worked, but it failed when the swimmer split into multiple foreground objects. An example of this is the women’s 50 m backstroke. In the first half of this race the swimmer in lane 8 regularly split into two or more foreground objects of similar area. This is shown in Figure 12.12. It led to jumps in the swimmer's coordinates of over a metre for certain sequences of frames and as a result this race had the worst agreement between manual and automated digitisation methods.

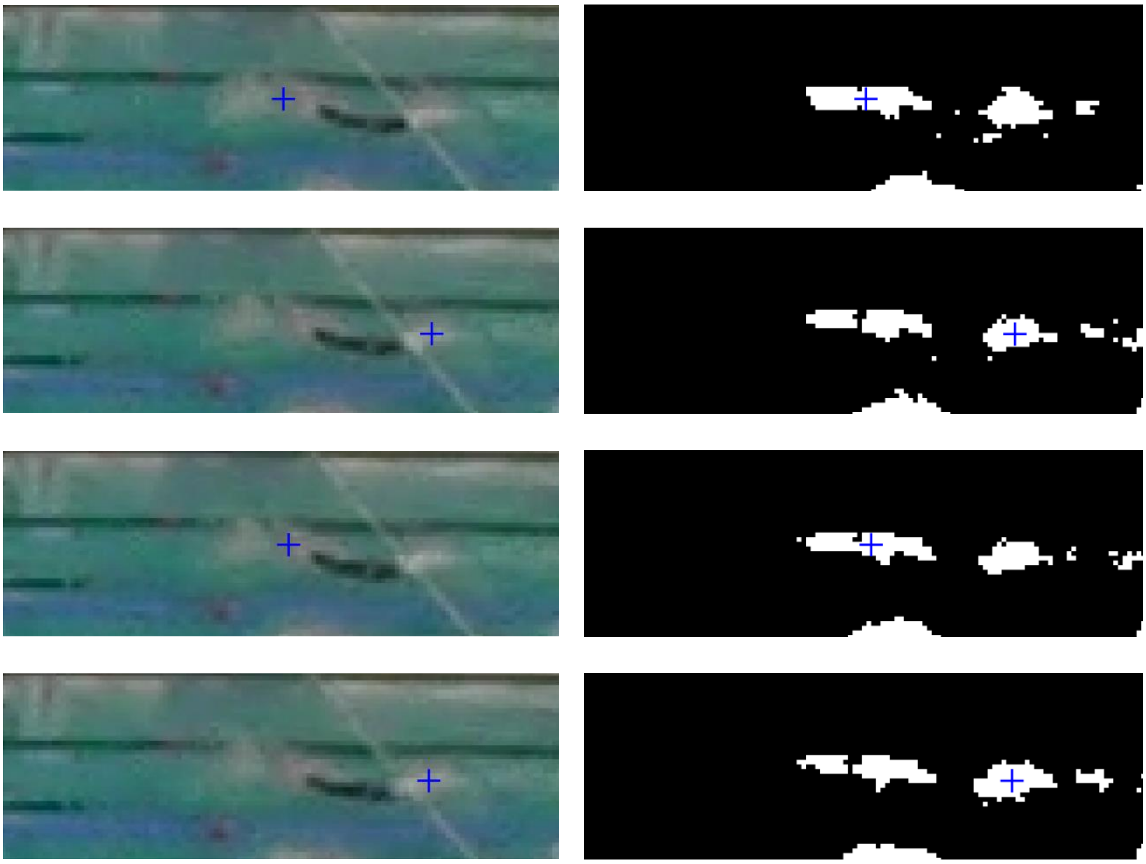


Figure 12.12. Example of high variability swimmer X coordinate from lane eight of the women's 50 m backstroke. This is a sequence of frames between 7.82 s and 7.88 s in the race. Blue crosses are the centroid of the foreground object with the largest area in pixels.

Two factors caused this fragmentation. First, the large visible area of swimsuit was not classified as a foreground object and this split the swimmer into at least two objects. This may have been a consequence of using a subtraction rather than an absolute difference when creating the difference image. This problem was not identified during development as the training set contained only male swimmers. Secondly, in general, the swimmers in the women's 50 m backstroke race generated equal quantities of splash with their arms and legs. This meant that the foreground objects that were their arms and legs were approximately equal in size. As a result, two or more equal sized foreground objects were created in the thresholded image and small inter-frame changes in area of these led to high variability swimmer coordinates.

Contrast this with the men’s 100 m breaststroke race. This race had the best agreement between manual and automated digitisation methods. The men wore swimsuits that extended from hip to knee and, due to the nature of the stroke, this swimsuit tended to stay below the water surface. This meant that the swimmer foreground object tended not to split at the swimsuit. As a result, the swimmers in this race were, in general, a single foreground object (as shown in Figure 12.13) and this led to low variability swimmer coordinates.

Figure 12.13. Extract from a thresholded image from the men’s 100 m breaststroke race. In this race, the swimmers tended to be a single foreground object.

12.4.6 Execution time

It took less time to automatically digitise a swimmer than it did to manually digitise one. The times in Table 12.3 for the automated method were for the digitisation of all eight swimmers in a lap whereas the manual times were for one swimmer per lap. So, to compare digitisation time for the two methods, the manual times would have to be multiplied by eight. Such multiplication showed that the automated method was between three and fifteen times faster than manual.

12.4.7 Agreement between methods

In general, as shown in Table 12.4, the mean speeds calculated by the automated and manual digitisation methods agreed. The 95% limits of agreement for the test set was -0.024 m s^{-1} to 0.034 m s^{-1}. As this range spanned zero, there was no evidence of a large systematic difference between the two methods, but there were random differences. Many factors could have led to this. For example:

- Digitisation target. The manual method digitised the centre of the swimmer's head whereas the automated one used the centroid of a digital image processing entity.
- Out-of-plane. In the manual method, the digitised points were on the calibrated plane and so would not be subject to out-of-plane errors. With the automated method, the swimmer's position was the defined as the centroid of a foreground object. In some instances, for example in breaststroke, much of the object could be well out-of-plane.
- Number of digitised frames. The automated method digitised the swimmer in every frame in the clean swimming phase. In contrast, the manual one used different digitisation strategies for different strokes. Also, in some frames, it was not possible to manually digitise the

swimmer (due to e.g. splash obscuring the swimmer's head); automated did not have this problem.

- Error in either method. A source of error in the automated method was identified above. There may also have been errors in the manual digitisation, e.g. incorrect alignment of the digitisation cursor.
- Non-linear speed. As shown in Chapter 9, it is likely that a swimmer does not maintain a constant speed in a lap. Yet, in the current chapter, two speeds calculated using simple linear regression was compared. A polynomial regression may have reduced the difference between the two methods.

12.5 Conclusion

This chapter developed an automated digitisation system and compared its results with those from the manual methods used in Chapter 9 and found:

- A sequence of five digital image processing steps was used to automatically digitise swimmers in video footage captured from a single-camera viewpoint.
- The highly dynamic nature of a swimming pool was successfully modelled using a temporal median filter.
- The swimmer model was simple: they were the largest connected region of pixels that were above an automatically selected threshold level.
- No large systematic differences between automated and manual digitisation methods of calculating speed.
- The automated digitisation system was between three and fifteen times faster than the manual method.

13 Summary

13.1 Introduction

This chapter summarises the work that has been described in this document. It reviews the errors in reconstructed distances that were investigated. The manual and automated digitisation of swimmers in the clean swimming phase of a race is also discussed. Finally, the chapter concludes with a discussion of possible future work.

13.2 Summary of work

13.2.1 Errors in reconstructed distances

The errors in reconstructed distances illustrated in Figure 13.1 were investigated.

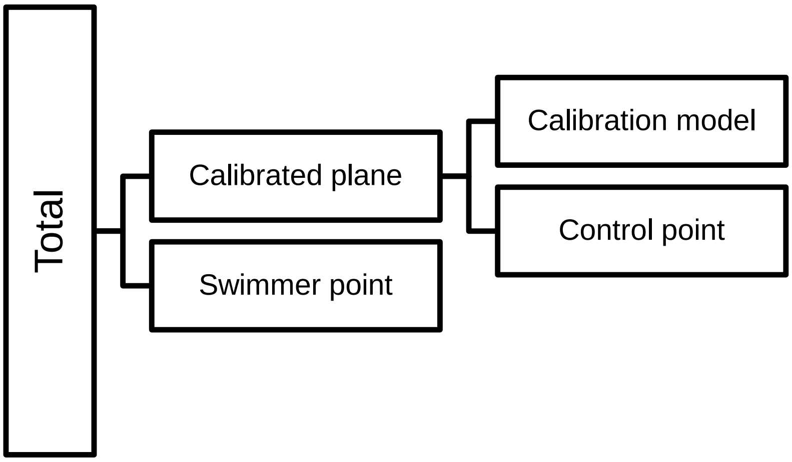


Figure 13.1. Errors in reconstructed distances that were investigated.

- Calibration model error was caused by a calibration model not having the correct relationship between distances in the pixel and world coordinate systems. A scale swimming pool was used to quantify this error. It was quantified for different viewpoints and different 2D-DLT and nonlinear calibration models. The calibration model error for the single-camera viewpoints was lower than that reported in the literature for a multi-camera system. In addition, it was found that a correctly chosen nonlinear calibration model always produced lower calibration model error than did a 2D-DLT one.

- Control point error showed how uncertainty in the measurement of the control points used to calculate a nonlinear calibration model propagated to error in reconstructed distances. A computer simulation was used to quantify this. Measurements of a swimming pool using a low-cost laser distance measurement device were deemed fit for purpose, but control point digitisation was identified as an area for improvement.
- Control point digitisation was improved by way of the line-line method. Six or more points on the straight lines that bounded a swimming pool were digitised. Simple linear regression was then used to calculate the pixel coordinates of the control points, which were at the intersections between pairs of lines. This method led to an improvement in precision compared to digitising the control point. The improvement was between 26% and 63%, dependent on viewpoint.
- This improvement in precision led to a reduction in control point error of between 27% and 63%, dependent on viewpoint. It was found that the largest average control point errors were in the distances at the start and end of each lap. As a result, measurements of distance in the clean swimming phase would be less affected than those in the other phases.
- Calibrated plane error was the combination of calibration model and control point errors. It showed how sensitive a calibration model was to uncertainties in the measurement of the control points that it was

calculated from. A computer simulation showed that nonlinear calibration models were reasonably insensitive to such uncertainties.

- Swimmer point error was the error in reconstructing a distance that was caused by uncertainty in the digitisation of a swimmer. A computer simulation was used to quantify this. The swimmer point error for the single-camera viewpoints was similar to that reported in the literature for a multi-camera system. Also, swimmer point error would have a smaller impact on the clean swimming phase than it would on the other race phases.
- Total error was the combination of all other errors. It showed how errors in the construction of a calibrated plane and digitisation of swimmers on that plane propagated to error in reconstructed distances. This error was found to be a straightforward combination of its constituent errors: calibrated plane and swimmer point. Calibrated plane error determined the bias and swimmer point the variability.
- How total error propagated to error in measurement of a swimmer’s speed in the clean swimming phase was assessed. This was done by calculating whether an enhancement of 1% in speed could be identified given the likely error in its calculation. An enhancement could not be identified for a non-perpendicular viewpoint, but it could for all the perpendicular ones.
- The following recommendations were made for a single-camera viewpoint for competition performance analysis: if the swimming competition venue allowed it, the best viewpoint was perpendicular and non-fisheye; a perpendicular and fisheye viewpoint should be chosen over a non-perpendicular and non-fisheye one.

13.2.2 Calculation of clean swimming speed

- Races at the British Gas Swimming Championships 2013 were filmed. The video footage was captured from a single-camera viewpoint, which used a fisheye lens converter to capture footage of the entire pool.
- The swimmers were manually digitised as often as possible in the clean swimming phase of each lap. This was time consuming: it typically took between five and ten minutes to digitise one lap for one swimmer. It gave an insight into speed within a lap; data such as this had not been previously reported in the literature.
- An automated digitisation method was developed. A simple sequence of digital image processing techniques was used to automatically calculate clean swimming speed for all swimmers in a race. This method was between three and fifteen times faster than the manual one.
- Overall, the mean speeds from the manual and automated digitisation methods were in good agreement, but the speed profiles less so.

13.3 Conclusions

The main findings are summarised below:

- The existing multi-camera competition performance analysis systems could be replaced with a single- camera one.
- If the competition venue allowed it, the camera’s viewpoint should be perpendicular and non-fisheye. This viewpoint could identify a 1% enhancement in speed for distances of 5 m or more.
- A perpendicular and fisheye viewpoint should be chosen over a non-perpendicular and non-fisheye one.

- An automated digitisation method has the potential to replace manual digitisation.

13.4 Future work

The following are possible areas for future study:

- 4K camcorder. A 4K camcorder captures video footage that is twice as wide and twice as high as the one used in this study. This would lead to an increase in the resolution, i.e. the number of pixels per metre. It is hypothesised that this would, in turn, lead to a decrease in error in reconstructed distances. The scale swimming pool and computer simulation developed in this study could be used to test this.
- More single camera viewpoints. In this study, four single camera viewpoints were assessed. One viewpoint not considered was a non-perpendicular and fisheye one. For these, the camera would be positioned away from the preferred, central location (i.e. not at 25 m in the X axis). Anecdotally, viewpoints such as these are available at most competition venues. The scale swimming pool and computer simulation developed in this study could be used to quantify error in reconstructed distances for these viewpoints.
- Use of advanced background model. The drowning prevention systems reviewed in Chapter 2 used, for example, block-based models of background objects, pixel motion frequency, and local search windows. Techniques such of these could enhance the automated digitisation system developed in Chapter 12. In particular, using a weighted averaging scheme to periodically update the clean background image,

rather than constructing a new one for each frame, should improve execution time.

- Use of advanced foreground model. A problem found in Chapter 12 was classifying the swimsuit as being part of a swimmer. Eng et al. (2003) and Lu and Tan (2004) modelled foreground objects by identifying clusters in colour space. This approach has the potential to identify a swimsuit, which tended to be a single colour. Another method to try would be a two-level threshold similar to the one used by Pogalin et al. (2007).
- Reduced frame rate for automated digitisation. The time required to process a race could be reduced by not processing every frame. More work is required to ascertain how a reduced frame rate would affect the accuracy and precision of the calculated speeds. The Bland and Altman (1986) method used in Chapter 12 could be used to assess this.
- Non-linear speed. Currently, speed is calculated using simple linear regression. As found in Chapter 9, it is likely that swimmers do not maintain a constant speed. So, it may be more appropriate to calculate speed with a polynomial regression.

14 References

ABDEL-AZIZ, Y. I. and KARARA, H. M. (1971). Direct linear transformation from comparator coordinates into object space coordinates in closer-range photogrammetry. In: *ASP symposium on close-range photogrammetry*, University of Illinois, 1971. American Society of Photogrammetry, Falls Church, VA, USA, 1-18.

AFORGE (2015). AForge.NET 2.2.5. [computer program]. AForge.

ALCOCK, Alison, HUNTER, Adam and BROWN, Nicholas (2009). Determination of football pitch locations from video footage and official pitch markings, *Sports Biomechanics*, **8** (2), 129-140.

ARELLANO, Raul, et al. (1994). Analysis of 50-, 100-, and 200-m freestyle swimmers at the 1992 Olympic Games. *Journal of Applied Biomechanics,* **10** (2), 189-199.

ARELLANO, Raul (2001). Modelling competitive swimming in different strokes and distances upon regression analysis: a study of the female participants of Sydney 2000 Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 21 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

BARTLETT, Roger (2014). *Introduction to sports biomechanics: analysing human movement patterns*. 3rd ed., Abingdon, Routledge.

BLAND, J. Martin and ALTMAN, Douglas G. (1986). Statistical methods for assessing agreement between two methods of clinical measurement. *The Lancet*, **327** (8476), 307-310.

BOUGUET (2013). *Camera calibration toolbox for MATLAB*. [online]. Last accessed 10 October 2014 at: http://www.vision.caltech.edu/bouguetj/calib_doc/.

BRADSKI, Gary and KAEHLER, Adrian (2008). *Learning OpenCV*. Sebastopol, O'Reilly.

BRITISH SWIMMING (2014). *Results and rankings*. [online]. Last accessed 23 September 2014 at: http://www.swimming.org/britishswimming/results-and-rankings.

CHAN, K. L. (2013). Detection of swimmer using dense optical flow motion map and intensity information. *Machine vision and applications*, **24** (1), 75-101.

CHATARD, Jean-Claude, et al. (2001a). Specific strategy for the medallists versus finalists and semi-finalists in the women's 200 m backstroke at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 11 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

CHATARD, Jean-Claude, et al. (2001b). Specific strategy for the medallists versus finalists and semi-finalists in the men's 200 m breaststroke at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 11 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

CHATARD, Jean-Claude, et al. (2001c). Specific strategy for the medallists versus finalists and semi-finalists in the women's 200 m breaststroke at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 11 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

CHATARD, Jean-Claude, et al. (2001d). Specific strategy for the medallists versus finalists and semi-finalists in the women's 200 m individual medley at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 14 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

CHATARD, Jean-Claude, et al. (2001e). Specific strategy for the medallists versus finalists and semi-finalists in the men's 200 m freestyle at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 15 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

CHENGALUR, Somadeepti N. and BROWN, Peter L. (1992). An analysis of male and female Olympic swimmers in the 200-meter events. *Canadian Journal of Sport Sciences,* **17** (2), 104-109.

CHOLLET, D., et al. (1996). Comparative analysis of 100 m and 200 m events in the four strokes in top level swimmers. *Journal of Human Movement Studies,* **31** (1), 25-38.

CHOLLET, Didier, et al. (1997). Stroking characteristic variations in the 100-m freestyle for male swimmers of differing skill. *Perceptual and Motor Skills,* **85** (1), 167-177.

CHOLLET, D. and PELAYO, P. (1999). Effects of different methodologies in calculating stroke length in swimming. *Journal of Human Movement Studies*, **36**, 127-136.

CHOLLET, Didier, TOURNY-CHOLLET, Claire, and HOGIE, Sylvain (2001). A kinematic study of butterfly turn in elite swimmers. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 19 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

CHOW, J. W. C., et al. (1984). Turning techniques of elite swimmers. *Journal of Sports Sciences,* **2** (3), 241-255.

COHEN, J. (1992). A power primer. *Psychological bulletin*, **112** (1), 155-159.

COSSOR, Jodi M. and MASON, Bruce R. (2001). Swim start performances at the Sydney 2000 Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 18 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

CRAIG, Albert B. and PENDERGAST, David R. (1979). Relationships of stroke rate, distance per stroke, and velocity in competitive swimming. *Medicine and Science in Sports,* **11** (3), 278-283.

CRAIG, Albert B., et al. (1985). Velocity, stroke rate, and distance per stroke during elite swimming competition. *Medicine and Science in Sports and Exercise,* **17** (6), 625-634.

Dartfish (2015). [online]. Last accessed 13 February 2015 at: http://www.dartfish.com/.

DRISCOLL, Heather and KELLEY, John (2013). *iSwim documentation: overview of software (beta version)*. Unpublished.

DUNN, Marcus et al. (2012). Reconstructing 2D planar coordinates using linear and nonlinear techniques. In: *30th International Conference on Biomechanics in Sports*, Melbourne, July 2-6 2012. Last accessed 20 October 2014 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2012.

EAST, David J. (1970). Swimming: An analysis of stroke frequency, stroke length and performance. *New Zealand Journal of Health, Physical Education and Recreation,* **3**, 16-27.

ENG et al. (2003). An automatic drowning detection surveillance system for challenging outdoor pool environments. In: *Proceedings of the 9th IEEE International Conference on Computer Vision*, Nice 2003. IEEE, 532-539.

FINA (2013a). *FINA facilities rules: 2013 – 2017*. [online]. Last accessed 25 July 2014 at: http://www.fina.org/H2O/index.php?option=com_content&view=article&id=4161&Itemid=184.

FINA (2013b). *FINA swimming rules: 2013 – 2017*. [online]. Last accessed 25 July 2014 at: http://www.fina.org/H2O/index.php?option=com_content&view=section&id=17&Itemid=184.

FINA (2015a). *FINA sports – history.* [online]. Last accessed 12 February 2015 at: http://www.fina.org/H2O/index.php?option=com_content&view=section&layout=blog&id=37&Itemid=357.

FINA (2015b). *Swimming world rankings.* [online]. Last accessed 24 February 2015 at: http://www.fina.org/H2O/index.php?option=com_wrapper&view=wrapper&Itemid=805.

GIROLD, Sebastien, et al. (2001a). Specific strategy for the medallists versus finalists and semi-finalists in the men's 200 m backstroke at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 14 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

GIROLD, Sebastien, et al. (2001b). Specific strategy for the medallists versus finalists and semi-finalists in the women's 200 m freestyle at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 15 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

HEIKKILA, J. and SILVEN, O. (1997). A four-step camera calibration procedure with implicit image correction. In: *Computer Vision and Pattern Recognition: proceedings of the IEEE Computer Society Conference on*, San Juan, Puerto Rico, June 17 – 19 1997. IEEE Computer Society Press, 1106-1112.

HELLARD, P., et al. (2008). Kinematic measures and stroke rate variability in elite female 200-m swimmers in the four swimming techniques: Athens 2004 Olympic semi-finalists and French National 2004 Championship semi-finalists. *Journal of Sports Sciences*, **26** (1), 35-46.

HUOT-MARCHAND, Francois, et al. (2005). Variations of stroking parameters associated with 200 m competitive performance improvement in top-standard front crawl swimmers. *Sports Biomechanics*, **4** (1), 89-100.

IKUTA, Yasushi, MASON, Bruce R., and COSSOR, Jodi. (2001). A comparison of Japanese finalists in the 100 m swimming races at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 16 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

INTERNATIONAL OLYMPIC COMMITTEE (2015). *Swimming equipment and history*. [online]. Last accessed 12 February 2015 at: http://www.olympic.org/swimming-equipment-and-history?tab=history.

INTERNATIONAL SOCIETY OF BIOMECHANICS (2014). *Movement Analysis Software*. [online]. Last accessed 04 September 2014 at: http://isbweb.org/software/movanal.html.

INTERNATIONAL TENNIS FEDERATION (2015). Dimensions (ITF CS 04/02). [online]. Last accessed 31 January 2015 at: http://www.itftennis.com/technical/courts/court-testing/dimensions.aspx.

JVC (2014). *GY-DV500 DV camcorder*. [online]. Last accessed 13 November 2014 at: http://pro.jvc.com/pro/attributes/prodv/brochure/gydv500.pdf.

KANNALA, Juho and BRANDT, Sami S. (2006). A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28** (8), 1335-1340.

KENNEDY, Patrick, et al. (1990). Analysis of male and female Olympic swimmers in the 100-meter events. *International Journal of Sport Biomechanics*, **6** (2), 187-195.

KWON, Young-Hoo (1999). Object plane deformation due to refraction in two-dimensional underwater motion analysis. *Journal of Applied Biomechanics*, **15**, 396-403.

LEICA GEOSYSTEMS (2014). Leica DISTO(TM) D2. [online]. Last accessed 19 August 2014 at: http://www.leica-geosystems.com/en/Leica-DISTO-D2_69656.htm.

LU, Wenmiao and TAN, Yap-Peng Tan (2004). A vision-based approach to early detection of drowning incidents in swimming pools. *IEEE transactions on circuits and systems for video technology*, **14** (2), 159-178.

MAGLISCHO, Ernest W. (2003). *Swimming fastest*. Champaign, Illinois, Human Kinetics.

MASON, Bruce, R., LOSCHNER, Constance, and FOWLIE, Jim (1995). Competition analysis at the World Championships. In: *13 International Symposium on Biomechanics in Sports*, Thunder Bay, July 18-22 1995. Last accessed 7 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS1995.

MASON, Bruce and COSSOR, Jodi (2000). What can we learn from competition analysis at the 1999 Pan Pacific swimming championships? In: *18 International Symposium on Biomechanics in Sports*, Hong Kong, June 25-30 2000. Last accessed 9 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2000.

MASON, Bruce R. and COSSOR, Jodi M. (2001). Swim turn performances at the Sydney Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 11 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

MATHWORKS (2014). *MATLAB 2014a*. [computer program]. MathWorks.

McLEAN, S. P. et al. (2004). *Factors affecting the accuracy of 2D-DLT calibration*. Paper presented at the 28th Annual Meeting of the American Society of Biomechanics, Portland (OR).

MICROSOFT (2010). Access 2010. [computer program]. Microsoft.

MICROSOFT (2012). .NET Framework 4.5. [computer program]. Microsoft.

MILLER, John A., HAY, James G. and WILSON, Barry D. (1984). Starting techniques of elite swimmers. *Journal of Sports Sciences,* **2** (3), 213-223.

OMEGA TIMING (2015). *Results – AQUATICS*. [online]. Last accessed 28 January 2015 at: http://www.omegatiming.com/Sport?sport=AQ.

OpenCV. (2015). [online]. Last accessed 30 January 2015 at: http://opencv.org/.

OTSU, Nobuyuki (1979). A threshold selection method from gray-level histograms. IEEE Transactions on systems, man, and cybernetics, **SMC-9** (1), 62-66.

PAI, Yi-Chung, HAY, James G. and WILSON, Barry D. (1984). Stroking techniques of elite swimmers. *Journal of Sports Sciences,* **2** (3), 225-239.

PAYTON, Carl J. (2008). Motion analysis using video. In: PAYTON, Carl J. and BARTLETT Roger M. (eds.). *Biomechanical evaluation of movement in sport and exercise: the British Association of Sport and Exercise Sciences guidelines*. London, Routledge, 8-32.

PELAYO, Patrick, et al. (1996). Stroking characteristics in freestyle swimming and relationships with anthropometric characteristics. *Journal of Applied Biomechanics,* **12** (2), 197-206.

PELAYO, Patrick, et al. (1996). Stroking characteristics in freestyle swimming and relationships with anthropometric characteristics. *Journal of Applied Biomechanics,* **12** (2), 197-206.

POGALIN, E., et al. (2007). Video-based training registration for swimmers. *International Journal of Computer Science in Sport*, **6** (1), 4-17.

PYNE, David, B., TREWIN, Cassie, B., and HOPKINS, William, G. (2004). Progression and variability of competitive performance of Olympic swimmers. *Journal of Sports Sciences*, **22** (7), 613-620.

RIEWALD, Scott (2001). Assessment of normalized distance per stroke and swimming efficiency in the 2000 Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 20 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

RUSS, John C. (2011). *The image processing handbook*. 6th ed., Boca Raton, CRC Press.

SANDERS, Russ, CAPPAERT, Jane M. and PEASE, David L. (1996). Technique characteristics of elite breaststroke swimmers. In: *14 International Symposium on Biomechanics in Sports*, Funchal, June 25-29 1996. Last accessed 8 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS1996.

SONY (2015). *HDR-PJ260VE*. [online]. Last accessed 22 January 2015 at: http://www.sony.co.uk/support/en/product/HDR-PJ260VE.

TAKAGI, Hideki, et al. (2004). Differences in stroke phases, arm-leg coordination, and velocity fluctuations due to event, gender and performance level in breaststroke. *Sports Biomechanics*, **3** (1), 15-27.

THOMPSON, K. G., et al. (2000). The relative importance of selected kinematic variables in relation to swimming performance in elite male and elite female 100m and 200m breaststroke swimmers. *Journal of Human Movement Studies,* **39** (1), 15-32.

THOMPSON, K.G., HALJAND, R. and MACLAREN, D.P. (2000). An analysis of selected kinematic variables in national and elite male and female 100-m and 200-m breaststroke swimmers. *Journal of Sports Sciences,* **18** (6), 421-431.

TOURNY-CHOLLET, C. et al. (2002). Kinematic analysis of butterfly turns of international and national swimmers. *Journal of Sports Sciences*, **20** (5), 383-390.

TSAI, Roger, Y. (1987). A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, **3** (4), 323-344.

VEIGA, Santiago, et al. (2010). The validity and reliability of a procedure for competition analysis in swimming based on individual distance measurements. In: *Proceedings of the XI International Symposium for Biomechanics and Medicine in Swimming*, Oslo 2010. Norwegian School of Sport Science, 182-184.

VEIGA, Santiago, et al. (2012). A new procedure for race analysis in swimming based on individual distance measures. *Journal of Sports Sciences*, **31** (2), 159-165.

WAKAYOSHI, K., et al. (1992). Analysis of swimming races in the 1989 Pan Pacific swimming championships and 1988 Japanese Olympic trials. In: *Sixth International Symposium on Biomechanics and Medicine in Swimming,* Liverpool 1990. Cambridge, E & FN Spon, 135-141.

WALTON, J. S. (1981). *Close-range cine-photogrammetry: A generalized technique for quantifying gross human motion*. PhD, Pennsylvania State University.

WANG, J. et al. (2004). Integrating color and motion to enhance human detection within aquatic environment. In: *IEEE International Conference on Multimedia and Expo, 2004*, Taipei, June 27 – 30 2004. IEEE, 1179-1182.

WILSON, Barry, et al. (2001). Relationships between stroke efficiency measures and freestyle swimming performance: an analysis of freestyle swimming events at the Sydney 2000 Olympic Games. In: *19 International Symposium on Biomechanics in Sports*, San Francisco, June 20-26 2001. Last accessed 11 February 2013 at: https://ojs.ub.uni-konstanz.de/cpa/issue/view/ISBS2001.

ZHANG, Z. (2000). *A flexible new technique for camera calibration*. IEEE transactions on pattern analysis and machine intelligence, **22** (11), 1330-1334.

15 Appendix

15.1 Calibration model error

This section contains reprojection error and calibration model error for each calibration model from Chapter 3. The columns |*k*|, Tangential, Principal point and Aspect ratio indicate the intrinsic parameters and distortion coefficients that were calculated in each calibration model.

15.1.1 V1

Procedure	|*k*|	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
DLT	N/A	N/A	N/A	N/A	N/A	1018.3
Nonlinear	0	N/A	No	No	3.80	96.8
Nonlinear	1	N/A	No	No	0.77	31.4
Nonlinear	2	N/A	No	No	0.72	21.4
Nonlinear	3	N/A	No	No	0.72	21.4
Nonlinear	4	N/A	No	No	0.72	21.7
Nonlinear	0	N/A	Yes	No	3.73	139.9
Nonlinear	1	N/A	Yes	No	0.60	16.0
Nonlinear	2	N/A	Yes	No	0.55	17.4
Nonlinear	3	N/A	Yes	No	0.55	17.4
Nonlinear	4	N/A	Yes	No	0.55	17.6
Nonlinear	0	N/A	No	Yes	3.42	108.6
Nonlinear	1	N/A	No	Yes	0.77	30.2
Nonlinear	2	N/A	No	Yes	0.72	19.6
Nonlinear	3	N/A	No	Yes	0.72	19.6
Nonlinear	4	N/A	No	Yes	0.72	19.9
Nonlinear	0	N/A	Yes	Yes	3.33	144.7
Nonlinear	1	N/A	Yes	Yes	0.61	16.6
Nonlinear	2	N/A	Yes	Yes	0.55	19.0
Nonlinear	3	N/A	Yes	Yes	0.55	19.0
Nonlinear	4	N/A	Yes	Yes	0.55	19.2

15.1.2 V2

Procedure	|*k*|	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
DLT	N/A	N/A	N/A	N/A	N/A	1353.5

Procedure	$\lvert k \rvert$	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
Nonlinear	0	N/A	No	No	15.61	537.4
Nonlinear	1	N/A	No	No	2.22	99.5
Nonlinear	2	N/A	No	No	0.88	31.3
Nonlinear	3	N/A	No	No	0.80	20.4
Nonlinear	4	N/A	No	No	0.80	20.5
Nonlinear	0	N/A	Yes	No	15.78	590.9
Nonlinear	1	N/A	Yes	No	2.11	83.6
Nonlinear	2	N/A	Yes	No	0.79	38.9
Nonlinear	3	N/A	Yes	No	0.69	31.9
Nonlinear	4	N/A	Yes	No	0.68	31.7
Nonlinear	0	N/A	No	Yes	14.15	588.0
Nonlinear	1	N/A	No	Yes	2.18	101.9
Nonlinear	2	N/A	No	Yes	0.88	32.2
Nonlinear	3	N/A	No	Yes	0.80	21.4
Nonlinear	4	N/A	No	Yes	0.80	21.5
Nonlinear	0	N/A	Yes	Yes	14.04	627.0
Nonlinear	1	N/A	Yes	Yes	2.08	84.4
Nonlinear	2	N/A	Yes	Yes	0.79	40.3
Nonlinear	3	N/A	Yes	Yes	0.69	33.1
Nonlinear	4	N/A	Yes	Yes	0.68	32.9

15.1.3 V3

Procedure	$\lvert k \rvert$	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
DLT	N/A	N/A	N/A	N/A	N/A	46.1
Nonlinear	0	No	No	No	0.58	41.0
Nonlinear	1	No	No	No	0.56	43.7
Nonlinear	2	No	No	No	0.55	44.6
Nonlinear	3	No	No	No	0.55	44.8
Nonlinear	0	Yes	No	No	0.51	39.3
Nonlinear	1	Yes	No	No	0.50	42.2
Nonlinear	2	Yes	No	No	0.48	43.7
Nonlinear	3	Yes	No	No	0.48	43.9
Nonlinear	0	No	Yes	No	0.52	42.3
Nonlinear	1	No	Yes	No	0.51	44.0
Nonlinear	2	No	Yes	No	0.50	44.8
Nonlinear	3	No	Yes	No	0.50	45.0
Nonlinear	0	Yes	Yes	No	0.50	38.9
Nonlinear	1	Yes	Yes	No	0.49	42.8
Nonlinear	2	Yes	Yes	No	0.47	44.9
Nonlinear	3	Yes	Yes	No	0.47	45.1
Nonlinear	0	No	No	Yes	0.56	41.9

Procedure	|*k*|	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
Nonlinear	1	No	No	Yes	0.56	43.2
Nonlinear	2	No	No	Yes	0.55	44.0
Nonlinear	3	No	No	Yes	0.54	44.1
Nonlinear	0	Yes	No	Yes	0.49	40.2
Nonlinear	1	Yes	No	Yes	0.49	41.5
Nonlinear	2	Yes	No	Yes	0.48	42.9
Nonlinear	3	Yes	No	Yes	0.48	43.1
Nonlinear	0	No	Yes	Yes	0.51	42.9
Nonlinear	1	No	Yes	Yes	0.51	43.7
Nonlinear	2	No	Yes	Yes	0.50	44.3
Nonlinear	3	No	Yes	Yes	0.50	44.5
Nonlinear	0	Yes	Yes	Yes	0.48	39.9
Nonlinear	1	Yes	Yes	Yes	0.48	41.8
Nonlinear	2	Yes	Yes	Yes	0.47	43.8
Nonlinear	3	Yes	Yes	Yes	0.46	44.0

15.1.4 V4

Procedure	|*k*|	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
DLT	N/A	N/A	N/A	N/A	N/A	10.3
Nonlinear	0	No	No	No	0.57	14.8
Nonlinear	1	No	No	No	0.54	13.1
Nonlinear	2	No	No	No	0.51	12.4
Nonlinear	3	No	No	No	0.51	12.3
Nonlinear	0	Yes	No	No	0.54	8.6
Nonlinear	1	Yes	No	No	0.49	5.5
Nonlinear	2	Yes	No	No	0.46	4.6
Nonlinear	3	Yes	No	No	0.46	4.5
Nonlinear	0	No	Yes	No	0.55	8.6
Nonlinear	1	No	Yes	No	0.49	6.2
Nonlinear	2	No	Yes	No	0.47	5.2
Nonlinear	3	No	Yes	No	0.47	5.2
Nonlinear	0	Yes	Yes	No	0.54	7.9
Nonlinear	1	Yes	Yes	No	0.49	5.9
Nonlinear	2	Yes	Yes	No	0.45	4.5
Nonlinear	3	Yes	Yes	No	0.45	4.3
Nonlinear	0	No	No	Yes	0.57	14.9
Nonlinear	1	No	No	Yes	0.51	14.0
Nonlinear	2	No	No	Yes	0.49	12.4
Nonlinear	3	No	No	Yes	0.48	12.2
Nonlinear	0	Yes	No	Yes	0.54	8.6
Nonlinear	1	Yes	No	Yes	0.47	7.6

Procedure	|*k*|	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
Nonlinear	2	Yes	No	Yes	0.44	5.3
Nonlinear	3	Yes	No	Yes	0.44	5.2
Nonlinear	0	No	Yes	Yes	0.55	8.5
Nonlinear	1	No	Yes	Yes	0.47	8.4
Nonlinear	2	No	Yes	Yes	0.45	6.3
Nonlinear	3	No	Yes	Yes	0.45	6.4
Nonlinear	0	Yes	Yes	Yes	0.54	7.8
Nonlinear	1	Yes	Yes	Yes	0.46	8.1
Nonlinear	2	Yes	Yes	Yes	0.43	5.6
Nonlinear	3	Yes	Yes	Yes	0.43	5.4

15.1.5 V5

Procedure	|*k*|	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
DLT	N/A	N/A	N/A	N/A	N/A	14.6
Nonlinear	0	No	No	No	0.60	13.6
Nonlinear	1	No	No	No	0.55	10.9
Nonlinear	2	No	No	No	0.52	7.7
Nonlinear	3	No	No	No	0.52	7.2
Nonlinear	0	Yes	No	No	0.52	13.1
Nonlinear	1	Yes	No	No	0.47	11.4
Nonlinear	2	Yes	No	No	0.45	8.8
Nonlinear	3	Yes	No	No	0.44	8.5
Nonlinear	0	No	Yes	No	0.55	14.1
Nonlinear	1	No	Yes	No	0.51	11.6
Nonlinear	2	No	Yes	No	0.48	8.2
Nonlinear	3	No	Yes	No	0.48	7.8
Nonlinear	0	Yes	Yes	No	0.48	11.2
Nonlinear	1	Yes	Yes	No	0.43	9.9
Nonlinear	2	Yes	Yes	No	0.40	8.0
Nonlinear	3	Yes	Yes	No	0.40	7.9
Nonlinear	0	No	No	Yes	0.56	13.6
Nonlinear	1	No	No	Yes	0.54	11.3
Nonlinear	2	No	No	Yes	0.52	7.9
Nonlinear	3	No	No	Yes	0.51	7.4
Nonlinear	0	Yes	No	Yes	0.49	13.4
Nonlinear	1	Yes	No	Yes	0.47	11.7
Nonlinear	2	Yes	No	Yes	0.44	9.0
Nonlinear	3	Yes	No	Yes	0.44	8.7
Nonlinear	0	No	Yes	Yes	0.53	14.3
Nonlinear	1	No	Yes	Yes	0.51	12.0
Nonlinear	2	No	Yes	Yes	0.48	8.5

Procedure	$\lvert k \rvert$	Tangential	Principal point	Aspect ratio	Reproj. error (pixels)	Cal. model error (10^{-3} m)
Nonlinear	3	No	Yes	Yes	0.48	8.0
Nonlinear	0	Yes	Yes	Yes	0.46	11.4
Nonlinear	1	Yes	Yes	Yes	0.43	10.1
Nonlinear	2	Yes	Yes	Yes	0.40	8.0
Nonlinear	3	Yes	Yes	Yes	0.39	7.8

15.2 Manual tracking speed

This section contains speeds that resulted from the manual tracking described in Chapter 9. Position is the order in which swimmers finished the race and n is the number of swimmer points that were used to calculate speed.

15.2.1 Men 100 m Breaststroke

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	28.72	6	18	1.581
1	2	61.10	6	22	1.490
2	1	28.89	4	22	1.606
2	2	60.99	4	26	1.531
3	1	28.33	7	23	1.624
3	2	61.35	7	28	1.494
4	1	28.45	1	20	1.645
4	2	59.80	1	24	1.552
5	1	28.14	3	20	1.658
5	2	60.11	3	24	1.562
6	1	28.36	2	19	1.622
6	2	60.06	2	23	1.531
7	1	28.87	5	23	1.628
7	2	61.01	5	26	1.535
8	1	28.68	8	18	1.604
8	2	61.39	8	24	1.495

15.2.2 Men 100 m Butterfly

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	24.76	7	20	1.867
1	2	53.77	7	23	1.690
2	1	24.58	6	19	1.900
2	2	53.07	6	22	1.725
3	1	24.19	3	18	1.911
3	2	52.56	3	20	1.744
4	1	24.49	1	18	1.904
4	2	51.97	1	20	1.772
5	1	24.73	4	18	1.880
5	2	52.76	4	20	1.768
6	1	24.52	2	18	1.884
6	2	52.40	2	20	1.761
7	1	24.82	5	17	1.864
7	2	52.78	5	18	1.737

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
8	1	25.27	8	19	1.813
8	2	54.09	8	20	1.704

15.2.3 Men 100 m Freestyle

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	24.22	7	88	2.037
1	2	50.96	7	83	1.856
2	1	23.90	5	180	2.046
2	2	50.20	5	300	1.875
3	1	23.57	3	83	2.051
3	2	49.48	3	25	1.937
4	1	22.97	1	92	2.114
4	2	48.67	1	264	1.903
5	1	23.91	2	140	2.016
5	2	49.17	2	210	1.941
6	1	23.89	4	155	2.002
6	2	50.13	4	301	1.879
7	1	24.20	6	115	1.990
7	2	50.28	6	37	1.867
8	1	24.51	8	228	2.010
8	2	51.02	8	154	1.847

15.2.4 Men 200 m Backstroke

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	27.88	5	333	1.725
1	2	57.72	5	407	1.631
1	3	88.39	5	610	1.582
1	4	119.52	5	570	1.535
2	1	27.64	6	329	1.715
2	2	58.06	6	351	1.581
2	3	89.48	6	397	1.532
2	4	120.48	6	365	1.534
3	1	27.77	4	349	1.700
3	2	57.93	4	565	1.590
3	3	87.97	4	463	1.591
3	4	118.77	4	619	1.541
4	1	27.29	2	391	1.733
4	2	56.86	2	498	1.657
4	3	86.97	2	483	1.618
4	4	117.20	2	531	1.609
5	1	28.22	1	501	1.711

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
5	2	58.44	1	548	1.606
5	3	87.66	1	466	1.688
5	4	116.36	1	413	1.700
6	1	27.76	8	185	1.746
6	2	58.08	8	259	1.619
6	3	89.58	8	238	1.562
6	4	121.20	8	324	1.532
7	1	27.05	3	56	1.748
7	2	56.35	3	54	1.661
7	3	86.38	3	218	1.630
7	4	117.23	3	135	1.556
8	1	27.60	7	108	1.763
8	2	58.44	7	172	1.607
8	3	89.60	7	175	1.583
8	4	120.67	7	326	1.546

15.2.5 Men 400 m Individual Medley

Due to a combination of factors it was not possible to track the swimmer in lane 8 on lap 7.

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	27.62	7	19	1.685
1	2	59.23	7	22	1.566
1	3	93.53	7	850	1.458
1	4	127.39	7	730	1.420
1	5	164.05	7	18	1.328
1	6	201.52	7	19	1.296
1	7	232.79	7	404	1.617
1	8	263.33	7	423	1.593
2	1	27.75	6	20	1.705
2	2	59.19	6	22	1.580
2	3	92.12	6	787	1.533
2	4	124.68	6	869	1.493
2	5	161.23	6	19	1.345
2	6	198.30	6	20	1.305
2	7	229.80	6	602	1.620
2	8	259.19	6	326	1.661
3	1	27.35	3	19	1.695
3	2	57.75	3	22	1.631
3	3	91.06	3	400	1.510
3	4	123.76	3	427	1.486
3	5	159.55	3	19	1.355

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
3	6	195.91	3	20	1.339
3	7	226.47	3	374	1.664
3	8	255.51	3	138	1.662
4	1	27.07	2	17	1.735
4	2	57.44	2	21	1.613
4	3	91.06	2	543	1.495
4	4	123.82	2	554	1.473
4	5	159.15	2	18	1.382
4	6	194.80	2	20	1.355
4	7	225.36	2	406	1.667
4	8	255.22	2	197	1.621
5	1	26.72	1	18	1.725
5	2	57.36	1	20	1.619
5	3	90.19	1	284	1.504
5	4	123.33	1	386	1.458
5	5	158.30	1	16	1.351
5	6	194.51	1	18	1.341
5	7	224.58	1	400	1.683
5	8	253.67	1	116	1.656
6	1	27.25	4	20	1.752
6	2	58.01	4	23	1.637
6	3	90.78	4	71	1.551
6	4	123.19	4	86	1.511
6	5	160.38	4	17	1.323
6	6	198.16	4	17	1.309
6	7	227.99	4	168	1.738
6	8	257.31	4	25	1.671
7	1	28.01	5	21	1.686
7	2	59.02	5	23	1.613
7	3	91.60	5	460	1.544
7	4	124.07	5	535	1.501
7	5	159.98	5	21	1.378
7	6	195.89	5	22	1.365
7	7	227.87	5	412	1.587
7	8	258.97	5	348	1.558
8	1	28.76	8	15	1.665
8	2	60.72	8	19	1.589
8	3	94.77	8	127	1.466
8	4	128.48	8	38	1.438
8	5	164.85	8	13	1.352
8	6	201.62	8	16	1.338
8	7	233.90	8	N/A	N/A
8	8	265.15	8	43	1.549

15.2.6 Women 50 m Backstroke

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	30.16	7	232	1.611
2	1	30.01	6	433	1.645
3	1	28.82	3	144	1.680
4	1	28.16	2	130	1.693
5	1	27.97	1	145	1.722
6	1	29.44	4	256	1.646
7	1	29.69	5	64	1.655
8	1	30.73	8	41	1.610

15.2.7 Women 50 m Butterfly

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	27.60	7	16	1.730
2	1	26.97	5	25	1.773
3	1	25.91	1	22	1.830
4	1	26.83	2	12	1.730
5	1	26.84	3	22	1.769
6	1	26.87	4	20	1.791
7	1	27.63	8	19	1.705
8	1	27.57	6	21	1.724

15.2.8 Women 100 m Freestyle

Due to a combination of factors it was not possible to track the swimmer in lane 7 on lap 1. The swimmer in lane 8 did not start the race.

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	26.89	7	286	1.821
1	2	57.05	7	474	1.633
2	1	26.98	6	403	1.802
2	2	55.93	6	314	1.692
3	1	26.53	3	389	1.835
3	2	55.59	3	524	1.700
4	1	26.53	1	257	1.813
4	2	54.82	1	394	1.737
5	1	26.83	2	203	1.802
5	2	55.30	2	402	1.736
6	1	26.31	4	90	1.824
6	2	55.82	4	316	1.686
7	1	26.90	4	N/A	N/A

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
7	2	55.82	4	64	1.717

15.2.9 Women 200 m Breaststroke

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	33.36	7	19	1.409
1	2	71.88	7	20	1.261
1	3	112.10	7	21	1.213
1	4	152.67	7	24	1.190
2	1	33.87	6	23	1.369
2	2	72.30	6	24	1.279
2	3	112.06	6	25	1.244
2	4	152.28	6	26	1.209
3	1	33.55	3	18	1.411
3	2	71.90	3	19	1.296
3	3	110.39	3	19	1.296
3	4	148.82	3	20	1.292
4	1	33.74	2	17	1.417
4	2	71.08	2	18	1.313
4	3	110.12	2	20	1.297
4	4	147.71	2	22	1.266
5	1	32.97	1	17	1.419
5	2	70.17	1	18	1.319
5	3	108.03	1	19	1.310
5	4	147.52	1	20	1.243
6	1	34.01	5	19	1.363
6	2	71.58	5	22	1.314
6	3	110.78	5	22	1.252
6	4	151.40	5	25	1.201
7	1	34.18	4	19	1.356
7	2	72.84	4	20	1.276
7	3	111.71	4	20	1.269
7	4	150.15	4	23	1.280
8	1	34.02	8	20	1.385
8	2	72.89	8	20	1.269
8	3	113.00	8	22	1.228
8	4	154.20	8	23	1.202

15.3 Total error

This section contains the total error described in Chapter 11.

Viewpoint	Test distance length (m)	n	Mean (10^{-3} m)	Std. dev. (10^{-3} m)
V1	5	560000	10.3	21.6
V1	10	500000	20.5	23.6
V1	15	440000	32.1	27.3
V1	20	380000	43.9	30.8
V1	25	320000	55.6	33.1
V1	30	260000	67.0	35.6
V1	35	200000	77.4	38.7
V1	40	140000	87.6	42.6
V1	45	80000	99.1	44.4
V1	50	40000	112.2	39.9
V2	5	560000	6.0	26.0
V2	10	500000	16.5	31.8
V2	15	440000	27.3	35.9
V2	20	380000	37.2	41.2
V2	25	320000	47.1	46.5
V2	30	260000	56.4	51.0
V2	35	200000	63.2	52.2
V2	40	140000	66.1	51.7
V2	45	80000	67.2	51.1
V2	50	40000	53.3	51.9
V3	5	560000	-6.0	56.2
V3	10	500000	-12.3	79.9
V3	15	440000	-13.9	104.8
V3	20	380000	-10.4	123.1
V3	25	320000	-7.2	138.3
V3	30	260000	-11.5	150.7
V3	35	200000	-24.6	155.5
V3	40	140000	-38.3	147.8
V3	45	80000	-39.9	124.8
V3	50	40000	-36.0	129.0
V4	5	560000	0.3	10.6
V4	10	500000	0.6	11.9
V4	15	440000	1.5	13.7
V4	20	380000	3.2	14.9
V4	25	320000	5.0	16.2
V4	30	260000	6.0	17.5
V4	35	200000	6.2	18.3
V4	40	140000	6.6	17.5
V4	45	80000	9.5	16.2
V4	50	40000	9.6	18.1

15.4 Speed error: two-point

This section contains the speed error described in Chapter 11. The two-point method was used to calculate speed.

15.4.1 Baseline speed (1.6 m s^{-1})

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V1	5	560000	0.003	0.007
V1	10	500000	0.003	0.004
V1	15	440000	0.003	0.003
V1	20	380000	0.004	0.002
V1	25	320000	0.004	0.002
V1	30	260000	0.004	0.002
V1	35	200000	0.004	0.002
V1	40	140000	0.004	0.002
V1	45	80000	0.004	0.002
V1	50	40000	0.004	0.001
V2	5	560000	0.002	0.008
V2	10	500000	0.003	0.005
V2	15	440000	0.003	0.004
V2	20	380000	0.003	0.003
V2	25	320000	0.003	0.003
V2	30	260000	0.003	0.003
V2	35	200000	0.003	0.002
V2	40	140000	0.003	0.002
V2	45	80000	0.002	0.002
V2	50	40000	0.002	0.002
V3	5	560000	-0.002	0.018
V3	10	500000	-0.002	0.013
V3	15	440000	-0.001	0.011
V3	20	380000	-0.001	0.010
V3	25	320000	0.000	0.009
V3	30	260000	-0.001	0.008
V3	35	200000	-0.001	0.007
V3	40	140000	-0.002	0.006
V3	45	80000	-0.001	0.004
V3	50	40000	-0.001	0.004
V4	5	560000	0.000	0.003
V4	10	500000	0.000	0.002
V4	15	440000	0.000	0.001
V4	20	380000	0.000	0.001
V4	25	320000	0.000	0.001
V4	30	260000	0.000	0.001
V4	35	200000	0.000	0.001

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V4	40	140000	0.000	0.001
V4	45	80000	0.000	0.001
V4	50	40000	0.000	0.001

15.4.2 Enhanced speed (1.616 m s^{-1})

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V1	5	560000	0.003	0.007
V1	10	500000	0.003	0.004
V1	15	440000	0.003	0.003
V1	20	380000	0.004	0.002
V1	25	320000	0.004	0.002
V1	30	260000	0.004	0.002
V1	35	200000	0.004	0.002
V1	40	140000	0.004	0.002
V1	45	80000	0.004	0.002
V1	50	40000	0.004	0.001
V2	5	560000	0.002	0.008
V2	10	500000	0.003	0.005
V2	15	440000	0.003	0.004
V2	20	380000	0.003	0.003
V2	25	320000	0.003	0.003
V2	30	260000	0.003	0.003
V2	35	200000	0.003	0.002
V2	40	140000	0.003	0.002
V2	45	80000	0.002	0.002
V2	50	40000	0.002	0.002
V3	5	560000	-0.002	0.018
V3	10	500000	-0.002	0.013
V3	15	440000	-0.001	0.011
V3	20	380000	-0.001	0.010
V3	25	320000	0.000	0.009
V3	30	260000	-0.001	0.008
V3	35	200000	-0.001	0.007
V3	40	140000	-0.002	0.006
V3	45	80000	-0.001	0.004
V3	50	40000	-0.001	0.004
V4	5	560000	0.000	0.003
V4	10	500000	0.000	0.002
V4	15	440000	0.000	0.001
V4	20	380000	0.000	0.001
V4	25	320000	0.000	0.001
V4	30	260000	0.000	0.001
V4	35	200000	0.000	0.001

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V4	40	140000	0.000	0.001
V4	45	80000	0.000	0.001
V4	50	40000	0.000	0.001

15.5 Speed error: simple linear regression

This section contains the speed error described in Chapter 11. The simple linear regression method was used to calculate speed.

15.5.1 Baseline speed (1.6 m s^{-1})

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V1	5	560000	0.003	0.007
V1	10	500000	0.003	0.004
V1	15	440000	0.003	0.003
V1	20	380000	0.004	0.002
V1	25	320000	0.004	0.002
V1	30	260000	0.004	0.002
V1	35	200000	0.004	0.002
V1	40	140000	0.004	0.002
V1	45	80000	0.004	0.002
V1	50	40000	0.004	0.001
V2	5	560000	0.002	0.008
V2	10	500000	0.003	0.005
V2	15	440000	0.003	0.004
V2	20	380000	0.003	0.003
V2	25	320000	0.003	0.003
V2	30	260000	0.003	0.002
V2	35	200000	0.003	0.002
V2	40	140000	0.003	0.002
V2	45	80000	0.003	0.001
V2	50	40000	0.003	0.001
V3	5	560000	-0.002	0.018
V3	10	500000	-0.002	0.013
V3	15	440000	-0.002	0.011
V3	20	380000	-0.001	0.010
V3	25	320000	0.000	0.009
V3	30	260000	0.000	0.009
V3	35	200000	0.000	0.008
V3	40	140000	-0.001	0.007
V3	45	80000	-0.001	0.005
V3	50	40000	-0.001	0.005
V4	5	560000	0.000	0.003
V4	10	500000	0.000	0.002
V4	15	440000	0.000	0.001
V4	20	380000	0.000	0.001
V4	25	320000	0.000	0.001
V4	30	260000	0.000	0.001
V4	35	200000	0.000	0.001

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V4	40	140000	0.000	0.001
V4	45	80000	0.000	0.001
V4	50	40000	0.000	0.001

15.5.2 Enhanced speed (1.616 m s^{-1})

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V1	5	560000	0.003	0.007
V1	10	500000	0.003	0.004
V1	15	440000	0.003	0.003
V1	20	380000	0.004	0.002
V1	25	320000	0.004	0.002
V1	30	260000	0.004	0.002
V1	35	200000	0.004	0.002
V1	40	140000	0.004	0.002
V1	45	80000	0.004	0.002
V1	50	40000	0.004	0.001
V2	5	560000	0.002	0.008
V2	10	500000	0.003	0.005
V2	15	440000	0.003	0.004
V2	20	380000	0.003	0.003
V2	25	320000	0.003	0.003
V2	30	260000	0.003	0.002
V2	35	200000	0.003	0.002
V2	40	140000	0.003	0.002
V2	45	80000	0.003	0.001
V2	50	40000	0.003	0.001
V3	5	560000	-0.002	0.018
V3	10	500000	-0.002	0.013
V3	15	440000	-0.002	0.011
V3	20	380000	-0.001	0.010
V3	25	320000	0.000	0.009
V3	30	260000	0.000	0.009
V3	35	200000	0.000	0.008
V3	40	140000	-0.001	0.007
V3	45	80000	-0.001	0.005
V3	50	40000	-0.001	0.005
V4	5	560000	0.000	0.003
V4	10	500000	0.000	0.002
V4	15	440000	0.000	0.001
V4	20	380000	0.000	0.001
V4	25	320000	0.000	0.001
V4	30	260000	0.000	0.001
V4	35	200000	0.000	0.001

Viewpoint	Test distance length (m)	n	Mean (m s^{-1})	Std. dev. (m s^{-1})
V4	40	140000	0.000	0.001
V4	45	80000	0.000	0.001
V4	50	40000	0.000	0.001

15.6 95% limits of agreement: two-point

This appendix shows the 95% limits of agreement for baseline (i.e. 1.6 m s^{-1}) and enhanced speed (i.e. 1.616 m s^{-1}) described in Chapter 11. Also shown is the gap between these ranges. The two-point method was used to calculate speed.

Viewpoint	Test distance length (m)	n	Baseline (m s^{-1})	Enhanced (m s^{-1})	Gap (m s^{-1})
V1	5	560000	1.589 to 1.617	1.605 to 1.633	-0.012
V1	10	500000	1.596 to 1.611	1.612 to 1.627	0.001
V1	15	440000	1.598 to 1.609	1.614 to 1.625	0.005
V1	20	380000	1.599 to 1.608	1.615 to 1.625	0.007
V1	25	320000	1.599 to 1.608	1.615 to 1.624	0.007
V1	30	260000	1.600 to 1.607	1.616 to 1.623	0.009
V1	35	200000	1.600 to 1.607	1.616 to 1.623	0.009
V1	40	140000	1.600 to 1.607	1.616 to 1.623	0.009
V1	45	80000	1.600 to 1.607	1.616 to 1.623	0.009
V1	50	40000	1.601 to 1.606	1.617 to 1.622	0.011
V2	5	560000	1.585 to 1.619	1.601 to 1.635	-0.018
V2	10	500000	1.592 to 1.613	1.608 to 1.629	-0.005
V2	15	440000	1.595 to 1.611	1.611 to 1.627	0.000
V2	20	380000	1.596 to 1.610	1.612 to 1.626	0.002
V2	25	320000	1.597 to 1.609	1.613 to 1.625	0.004
V2	30	260000	1.598 to 1.608	1.614 to 1.625	0.006
V2	35	200000	1.598 to 1.608	1.614 to 1.624	0.006
V2	40	140000	1.599 to 1.607	1.614 to 1.623	0.007
V2	45	80000	1.599 to 1.606	1.615 to 1.622	0.009
V2	50	40000	1.598 to 1.605	1.614 to 1.621	0.009
V3	5	560000	1.562 to 1.634	1.578 to 1.650	-0.056
V3	10	500000	1.572 to 1.624	1.588 to 1.640	-0.036
V3	15	440000	1.576 to 1.621	1.592 to 1.637	-0.029
V3	20	380000	1.579 to 1.619	1.595 to 1.635	-0.024
V3	25	320000	1.582 to 1.617	1.598 to 1.633	-0.019
V3	30	260000	1.583 to 1.615	1.599 to 1.632	-0.016
V3	35	200000	1.585 to 1.613	1.601 to 1.629	-0.012
V3	40	140000	1.587 to 1.610	1.603 to 1.626	-0.007
V3	45	80000	1.590 to 1.607	1.606 to 1.624	-0.001
V3	50	40000	1.591 to 1.607	1.606 to 1.623	-0.001
V4	5	560000	1.593 to 1.607	1.609 to 1.623	0.002
V4	10	500000	1.596 to 1.604	1.612 to 1.620	0.008
V4	15	440000	1.597 to 1.603	1.613 to 1.619	0.010
V4	20	380000	1.598 to 1.603	1.614 to 1.619	0.011
V4	25	320000	1.598 to 1.602	1.614 to 1.618	0.012

Viewpoint	Test distance length (m)	n	Baseline (m s^{-1})	Enhanced (m s^{-1})	Gap (m s^{-1})
V4	30	260000	1.598 to 1.602	1.614 to 1.618	0.012
V4	35	200000	1.599 to 1.602	1.615 to 1.618	0.013
V4	40	140000	1.599 to 1.602	1.615 to 1.618	0.013
V4	45	80000	1.599 to 1.601	1.615 to 1.618	0.014
V4	50	40000	1.599 to 1.601	1.615 to 1.617	0.014

15.7 95% limits of agreement: simple linear regression

This appendix shows the 95% limits of agreement for baseline (i.e. 1.6 m s^{-1}) and enhanced speed (i.e. 1.616 m s^{-1}) described in Chapter 11. Also shown is the gap between these ranges. The simple linear regression method was used to calculate speed.

Viewpoint	Test distance length (m)	n	Baseline (m s^{-1})	Enhanced (m s^{-1})	Gap (m s^{-1})
V1	5	560000	1.589 to 1.617	1.605 to 1.633	-0.012
V1	10	500000	1.596 to 1.611	1.612 to 1.627	0.001
V1	15	440000	1.598 to 1.609	1.614 to 1.625	0.005
V1	20	380000	1.599 to 1.608	1.615 to 1.624	0.006
V1	25	320000	1.599 to 1.608	1.615 to 1.624	0.008
V1	30	260000	1.600 to 1.607	1.616 to 1.623	0.009
V1	35	200000	1.600 to 1.607	1.616 to 1.623	0.009
V1	40	140000	1.600 to 1.607	1.616 to 1.623	0.010
V1	45	80000	1.601 to 1.607	1.617 to 1.623	0.010
V1	50	40000	1.601 to 1.606	1.617 to 1.622	0.011
V2	5	560000	1.585 to 1.619	1.601 to 1.635	-0.017
V2	10	500000	1.592 to 1.613	1.608 to 1.629	-0.004
V2	15	440000	1.595 to 1.610	1.611 to 1.627	0.001
V2	20	380000	1.597 to 1.609	1.613 to 1.625	0.004
V2	25	320000	1.598 to 1.609	1.614 to 1.625	0.005
V2	30	260000	1.598 to 1.608	1.614 to 1.624	0.006
V2	35	200000	1.599 to 1.608	1.615 to 1.624	0.007
V2	40	140000	1.599 to 1.607	1.615 to 1.623	0.009
V2	45	80000	1.600 to 1.605	1.616 to 1.621	0.011
V2	50	40000	1.600 to 1.605	1.616 to 1.621	0.011
V3	5	560000	1.562 to 1.634	1.578 to 1.650	-0.056
V3	10	500000	1.572 to 1.624	1.588 to 1.640	-0.035
V3	15	440000	1.576 to 1.621	1.592 to 1.637	-0.029
V3	20	380000	1.579 to 1.619	1.595 to 1.636	-0.025
V3	25	320000	1.581 to 1.618	1.597 to 1.634	-0.021
V3	30	260000	1.583 to 1.617	1.599 to 1.633	-0.018
V3	35	200000	1.584 to 1.615	1.600 to 1.631	-0.015
V3	40	140000	1.586 to 1.613	1.602 to 1.629	-0.012
V3	45	80000	1.589 to 1.610	1.605 to 1.626	-0.006
V3	50	40000	1.589 to 1.609	1.605 to 1.625	-0.004
V4	5	560000	1.593 to 1.607	1.609 to 1.623	0.002
V4	10	500000	1.596 to 1.604	1.612 to 1.620	0.008
V4	15	440000	1.597 to 1.603	1.613 to 1.619	0.010
V4	20	380000	1.598 to 1.603	1.614 to 1.619	0.011
V4	25	320000	1.598 to 1.602	1.614 to 1.618	0.012

Viewpoint	Test distance length (m)	n	Baseline (m s^{-1})	Enhanced (m s^{-1})	Gap (m s^{-1})
V4	30	260000	1.599 to 1.602	1.615 to 1.618	0.012
V4	35	200000	1.599 to 1.602	1.615 to 1.618	0.013
V4	40	140000	1.599 to 1.602	1.615 to 1.618	0.013
V4	45	80000	1.599 to 1.602	1.615 to 1.618	0.014
V4	50	40000	1.599 to 1.602	1.615 to 1.618	0.014

15.8 Automated tracking speed

This section contains speeds that resulted from the automated tracking described in Chapter 12. Position is the order in which swimmers finished the race and n is the number of swimmer points that were used to calculate speed.

15.8.1 Men 100 m Breaststroke

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	28.72	6	1059	1.580
1	2	61.10	6	1288	1.489
2	1	28.89	4	1177	1.604
2	2	60.99	4	1320	1.531
3	1	28.33	7	1097	1.621
3	2	61.35	7	1354	1.495
4	1	28.45	1	1075	1.643
4	2	59.80	1	1264	1.554
5	1	28.14	3	1113	1.650
5	2	60.11	3	1326	1.568
6	1	28.36	2	1049	1.620
6	2	60.06	2	1259	1.539
7	1	28.87	5	1088	1.629
7	2	61.01	5	1279	1.536
8	1	28.68	8	1042	1.607
8	2	61.39	8	1318	1.504

15.8.2 Men 100 m Butterfly

Lane	n	Split time (s)	Position	Lap	Speed (m s^{-1})
1	929	24.76	7	1	1.887
1	1134	53.77	7	2	1.731
2	930	24.58	6	1	1.877
2	1120	53.07	6	2	1.745
3	928	24.19	3	1	1.917
3	1089	52.56	3	2	1.741
4	908	24.49	1	1	1.932
4	996	51.97	1	2	1.795
5	938	24.73	4	1	1.864
5	1005	52.76	4	2	1.800
6	905	24.52	2	1	1.908
6	1021	52.40	2	2	1.786
7	887	24.82	5	1	1.871
7	991	52.78	5	2	1.729

Lane	n	Split time (s)	Position	Lap	Speed (m s^{-1})
8	981	25.27	8	1	1.820
8	1045	54.09	8	2	1.706

15.8.3 Men 100 m Freestyle

Lane	Lap	Split time (s)	Position	n	Speed (m s-1)
1	1	24.22	7	902	2.023
1	2	50.96	7	1027	1.850
2	1	23.90	5	777	2.049
2	2	50.20	5	1043	1.890
3	1	23.57	3	802	2.043
3	2	49.48	3	775	1.927
4	1	22.97	1	778	2.106
4	2	48.67	1	992	1.912
5	1	23.91	2	802	2.028
5	2	49.17	2	970	1.953
6	1	23.89	4	897	2.011
6	2	50.13	4	1024	1.917
7	1	24.20	6	798	2.010
7	2	50.28	6	1013	1.906
8	1	24.51	8	985	2.028
8	2	51.02	8	1005	1.856

15.8.4 Men 200 m Backstroke

Lane	Lap	Split time (s)	Position	n	Speed (m s-1)
1	1	27.88	5	847	1.726
1	2	57.72	5	992	1.634
1	3	88.39	5	1148	1.582
1	4	119.52	5	1110	1.521
2	1	27.64	6	877	1.724
2	2	58.06	6	858	1.572
2	3	89.48	6	1008	1.532
2	4	120.48	6	950	1.523
3	1	27.77	4	962	1.711
3	2	57.93	4	1042	1.586
3	3	87.97	4	1037	1.587
3	4	118.77	4	1083	1.536
4	1	27.29	2	886	1.744
4	2	56.86	2	913	1.659
4	3	86.97	2	1003	1.621
4	4	117.20	2	1071	1.608
5	1	28.22	1	866	1.721

Lane	Lap	Split time (s)	Position	n	Speed (m s-1)
5	2	58.44	1	1023	1.592
5	3	87.66	1	985	1.689
5	4	116.36	1	911	1.692
6	1	27.76	8	896	1.759
6	2	58.08	8	1062	1.624
6	3	89.58	8	1153	1.565
6	4	121.20	8	1183	1.541
7	1	27.05	3	708	1.791
7	2	56.35	3	871	1.682
7	3	86.38	3	871	1.660
7	4	117.23	3	880	1.562
8	1	27.60	7	864	1.796
8	2	58.44	7	989	1.613
8	3	89.60	7	1066	1.593
8	4	120.67	7	1141	1.525

15.8.5 Men 400 m Individual Medley

Due to a combination of factors it was not possible to manually track the swimmer in lane 8 on lap 7. As a result, automated tracking was not attempted for this lane and lap.

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	27.62	7	1084	1.690
1	2	59.23	7	1313	1.572
1	3	93.53	7	1337	1.464
1	4	127.39	7	1297	1.409
1	5	164.05	7	1466	1.325
1	6	201.52	7	1524	1.299
1	7	232.79	7	1302	1.622
1	8	263.33	7	1248	1.608
2	1	27.75	6	1097	1.708
2	2	59.19	6	1286	1.584
2	3	92.12	6	1212	1.526
2	4	124.68	6	1237	1.487
2	5	161.23	6	1454	1.348
2	6	198.30	6	1544	1.307
2	7	229.80	6	1327	1.619
2	8	259.19	6	1263	1.659
3	1	27.35	3	1030	1.706
3	2	57.75	3	1250	1.641
3	3	91.06	3	1236	1.512

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
3	4	123.76	3	1218	1.485
3	5	159.55	3	1386	1.358
3	6	195.91	3	1444	1.346
3	7	226.47	3	1270	1.652
3	8	255.51	3	1221	1.663
4	1	27.07	2	1024	1.726
4	2	57.44	2	1286	1.612
4	3	91.06	2	1263	1.500
4	4	123.82	2	1291	1.476
4	5	159.15	2	1421	1.386
4	6	194.80	2	1474	1.346
4	7	225.36	2	1269	1.661
4	8	255.22	2	1176	1.619
5	1	26.72	1	1019	1.733
5	2	57.36	1	1211	1.615
5	3	90.19	1	1174	1.523
5	4	123.33	1	1207	1.462
5	5	158.30	1	1344	1.357
5	6	194.51	1	1397	1.345
5	7	224.58	1	1219	1.681
5	8	253.67	1	1206	1.657
6	1	27.25	4	1089	1.764
6	2	58.01	4	1289	1.640
6	3	90.78	4	1157	1.570
6	4	123.19	4	1062	1.532
6	5	160.38	4	1258	1.330
6	6	198.16	4	1225	1.311
6	7	227.99	4	1067	1.732
6	8	257.31	4	968	1.657
7	1	28.01	5	1116	1.689
7	2	59.02	5	1276	1.607
7	3	91.60	5	1235	1.553
7	4	124.07	5	1187	1.499
7	5	159.98	5	1482	1.380
7	6	195.89	5	1483	1.370
7	7	227.87	5	1347	1.598
7	8	258.97	5	1267	1.573
8	1	28.76	8	855	1.674
8	2	60.72	8	1114	1.598
8	3	94.77	8	1059	1.472
8	4	128.48	8	885	1.457
8	5	164.85	8	1124	1.345
8	6	201.62	8	1359	1.336
8	7	233.90	8	N/A	N/A
8	8	265.15	8	1317	1.555

15.8.6 Women 50 m Backstroke

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	30.16	7	968	1.621
2	1	30.01	6	1007	1.664
3	1	28.82	3	790	1.671
4	1	28.16	2	801	1.737
5	1	27.97	1	829	1.722
6	1	29.44	4	994	1.658
7	1	29.69	5	808	1.679
8	1	30.73	8	830	1.679

15.8.7 Women 50 m Butterfly

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	27.60	7	844	1.729
2	1	26.97	5	1040	1.775
3	1	25.91	1	997	1.853
4	1	26.83	2	930	1.744
5	1	26.84	3	1007	1.781
6	1	26.87	4	934	1.813
7	1	27.63	8	934	1.701
8	1	27.57	6	1053	1.720

15.8.8 Women 100 m Freestyle

Due to a combination of factors it was not possible to manually track the swimmer in lane 7 on lap 1. As a result, automated tracking was not attempted for this lane and lap. The swimmer in lane 8 did not start the race.

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	26.89	7	1004	1.817
1	2	57.05	7	1189	1.645
2	1	26.98	6	1005	1.792
2	2	55.93	6	1113	1.702
3	1	26.53	3	1051	1.846
3	2	55.59	3	1242	1.722
4	1	26.53	1	1009	1.819
4	2	54.82	1	1104	1.743
5	1	26.83	2	1004	1.821
5	2	55.30	2	1189	1.742
6	1	26.31	4	933	1.836

6	2	55.82	4	1081	1.701
7	1	26.90	4	N/A	N/A
7	2	55.82	4	896	1.721

15.8.9 Women 200 m Breaststroke

Lane	Lap	Split time (s)	Position	n	Speed (m s^{-1})
1	1	33.36	7	1342	1.407
1	2	71.88	7	1620	1.260
1	3	112.10	7	1703	1.204
1	4	152.67	7	1775	1.183
2	1	33.87	6	1349	1.329
2	2	72.30	6	1561	1.264
2	3	112.06	6	1613	1.246
2	4	152.28	6	1664	1.214
3	1	33.55	3	1251	1.411
3	2	71.90	3	1502	1.297
3	3	110.39	3	1495	1.292
3	4	148.82	3	1476	1.293
4	1	33.74	2	1213	1.390
4	2	71.08	2	1521	1.314
4	3	110.12	2	1618	1.298
4	4	147.71	2	1652	1.268
5	1	32.97	1	1211	1.419
5	2	70.17	1	1422	1.318
5	3	108.03	1	1446	1.317
5	4	147.52	1	1549	1.245
6	1	34.01	5	1322	1.370
6	2	71.58	5	1497	1.316
6	3	110.78	5	1566	1.258
6	4	151.40	5	1724	1.185
7	1	34.18	4	1268	1.352
7	2	72.84	4	1470	1.278
7	3	111.71	4	1519	1.269
7	4	150.15	4	1526	1.284
8	1	34.02	8	1385	1.379
8	2	72.89	8	1614	1.268
8	3	113.00	8	1702	1.217
8	4	154.20	8	1707	1.194

15.9 Glossary of terms

Term	Description
Aspect ratio	Indicates whether the pixels in an image are square. A value of 1 indicates they are.
Background object	Everything in an image that is not a swimmer.
Barrel distortion	A type of radial distortion, which is common in wide-angle and fisheye lens.
Calibrated plane	A plane in the world coordinate system for which a calibration model is valid.
Calibration model	Calculated by a calibration procedure. Used to do projection and reconstruction.
Calibration object	An object used in the calculation of a calibration model.
Calibration procedure	Defines how to calculate a calibration model and the equations that the calibration model uses to do projection and reconstruction.
Control point	A point used in the calculation of a calibration model.
Coordinate digitiser	Used to digitise a swimmer in video footage.
Differencing	A digital image processing technique. Subtracts an image of the background from an image that contains background and foreground objects.
Digitisation	The process of obtaining the pixels coordinates of a swimmer from video footage.
Distortion coefficients	Describes two types of distortion: radial and tangential.
Extrinsic parameters	Two matrices (called *R* and *T*), which describe a camera's pose.
FHD	Full High Definition video footage. An image is 1920 pixels wide by 1080 pixels high.
Fixed distance analysis	Boundaries between race phases are at fixed distances in the swimming direction.
Foreground object	A swimmer in an image.
Frame or Image	A still image extracted from video footage. Consists of a two dimensional matrix of pixels.
Image plane	The plane in a camera onto which an image is projected.
Individual distance analysis	Boundaries between race phases are defined by a swimmer action, e.g. first hand entry after a period of underwater swimming.
Intrinsic parameters	Focal length, principal point and skew.
Lane ropes	Separate the lanes in a swimming pool. Consists of hundreds of coloured floats.
Optical axis	An imaginary line that is perpendicular to the image plane.
Pixel	A picture element.
Pixel coordinate system	Defines the index of a pixel in an image. The pixel coordinate system has U and V axes
Pool deck	The dry-land area that surrounds a swimming pool.

Term	Description
Principal point	Intersection of the optical axis and image plane. It is also the centre of the radial distortion.
Projection	The transformation from world to pixel coordinate systems.
Race phase	One of the four parts that an analysis divides a race into: a start, turns, a finish and clean swimming.
Race time	The time taken to complete a race.
Radial distortion	Results from differences in magnification across a lens.
RDE	Reconstructed Distance Error. The difference between the ground truth and reconstructed length of a test distance.
Reconstruction	The transformation from pixel to world coordinate systems.
Resolution	The number of pixels in an image that represent a metre.
RMSE	Root Mean Square Error. A measure of reconstruction accuracy.
ROI	A region of interest in an image of a swimming pool.
SD	Standard Definition video footage. An image is 720 pixels wide by 576 pixels high.
Skew	Angle between pixel coordinate system axes.
Tangential distortion	Caused by the misalignment of a camera's lens and image plane.
Test distance	A distance used to assess the accuracy of a reconstruction.
Test point	A point used to assess the accuracy of a reconstruction.
Thresholding	Classifies as background each pixel that is less than a certain value, which is called the threshold. The other pixels are part of a foreground object.
Viewpoint	Encapsulates the camera location, orientation, external lens and settings that were used to capture video footage of a swimming pool
World coordinate system	Represents 3D points and distances in the real world and has X, Y and Z axes.