Chapter 4

Measurement of the three-dimensional locus of moving targets

The previous chapters have emphasised the need to quantify the motion of vegetation elements in order to improve our understanding of the radar backscatter processes and of the coherence. The aim of this chapter is to present the method used in the research to measure the motion of vegetation in three dimensions. It will become clear in the next sections that the approach developed for the purposes of the research can be applied to more general cases where three-dimensional positions of moving targets are required. For that reason, the hardware and software presented here should be viewed as a package applicable to a wide range of scientific purposes. The concluding section of the chapter will come back to this aspect. Before that, section 4.1 introduces the design process leading to the implementation of the system. The detailed development of the different aspects of measurement system (hardware and software) are presented in sections 4.2 to 4.7. Section 4.8 shows how the system is used in practice for the measurement of wheat motion. It is worth emphasising here that the measurement system developed for the purposes of the research was designed and built from no previous experience about equivalent methods. The design of such a system, with the advantages listed in section 4.9, stands on its own as an original part of the research.

4.1 Design philosophy and process

The extraction of three-dimensional information from a pair of stereo images is not a new field of study. It was demonstrated with a pair of TV cameras in [58] as early as 1978. In [59] the mathematical basis and the methods required to track a moving object are described. More recently, stereo imaging has been the subject of a wide range of applications, including the generation of topographic maps from stereo images of the Earth surface acquired from space. Although the mathematical formulation of the stereo problem and of motion retrieval given in [58, 59] still applies, the availability of more modern hardware and the possibility to use much improved computing power have considerably increased the capabilities of current systems for stereo imaging and photogrammetry. The areas in constant development include object recognition ([60] provides a good overview of the current state of the art in this domain), and the development of low-cost photogrammetry systems [61].

The existing techniques for the retrieval of the three dimensional position of moving objects present three main limitations which make them unapplicable to the purposes of the research here. The first of these limitations is on cost. Most of the past stereo systems have been developed for industrial or commercial purposes and usually require expensive equipment, in particular concerning the cameras. The second limitation concerns the portability of the existing systems. A large proportion of them are used in laboratory environments where there is no need for a transportable measurement system. The subject of study is placed in the laboratory, as opposed to having a system capable of measurements in the natural environment of the subject. The third limitation is not general to all photogrammetry systems and applications: it concerns the repeatability in time of the measurements. Some applications require time consuming processing algorithms which limit the frequency at which the measurements can be repeated.

The design presented in the next sections is driven by the three aspects which impose limits on current systems: low-cost, transportability, and repeatability. In order to meet these three design drivers, a novel approach is necessary. The requirement to keep the cost of the entire system low imposes to use products of high consumption, where commercial competitiveness results in the lowest available prices. The transportability of the system is necessary if it is to be used outdoors, so it should be of minimum size and mass. The required repeatability imposes constraints on the processing techniques: they should be kept simple for better time efficiency. The design requirements above should not overshadow the primary requirement of the system, which is to provide accurate positions of a moving target for scientific uses. The accuracy of the position retrieval depends on the application. For the case dealt with in this research, the accuracy requirement was stated in section 3.6.

The basic starting idea of the design is that any system of several cameras pointing at different angles towards a single object is sufficient to retrieve the 3D position of this object. For 3D motion, video cameras are required. Starting from this basic idea, the issues to address concern:

- the number of video cameras required,
- the interfacing between the video cameras and a computer,
- the video data format usable for data processing,
- the recognition and tracking of the moving object in the video sequence,
- the definition of adequate reference systems and of the system geometry in order to relate the positions of the object in the video sequence to positions in a real world coordinate system.

It is the purpose of the following sections to answer these questions, keeping in mind the design drivers mentioned above. The remainder of this chapter is divided as follows:

- Section 4.2 presents the general geometry of the stereo system and the mathematical basis required to determine the coordinates of an object.
- Section 4.3 describes the calibration of the measurement system, i.e. the link between the positions of the cameras and a fixed reference system related to the scenery.
- Section 4.4 shows how the 3D coordinates of an object are calculated from its position in the video image.
- The hardware and software required for video data management are presented in section 4.5.
- The tracking algorithm is detailed in section 4.6, together with tests on its accuracy, robustness, and processing time efficiency.
- Section 4.7 deals with the calibration of the video image, i.e. the translation of image coordinates into coordinates related to the geometry defined in section 4.2.
- Finally section 4.8 shows how the system is used in practice for *in situ* measurements, and more specifically in the case of interest here, i.e. the measurement of wheat motion.

4.2 System geometry and basic model formulation

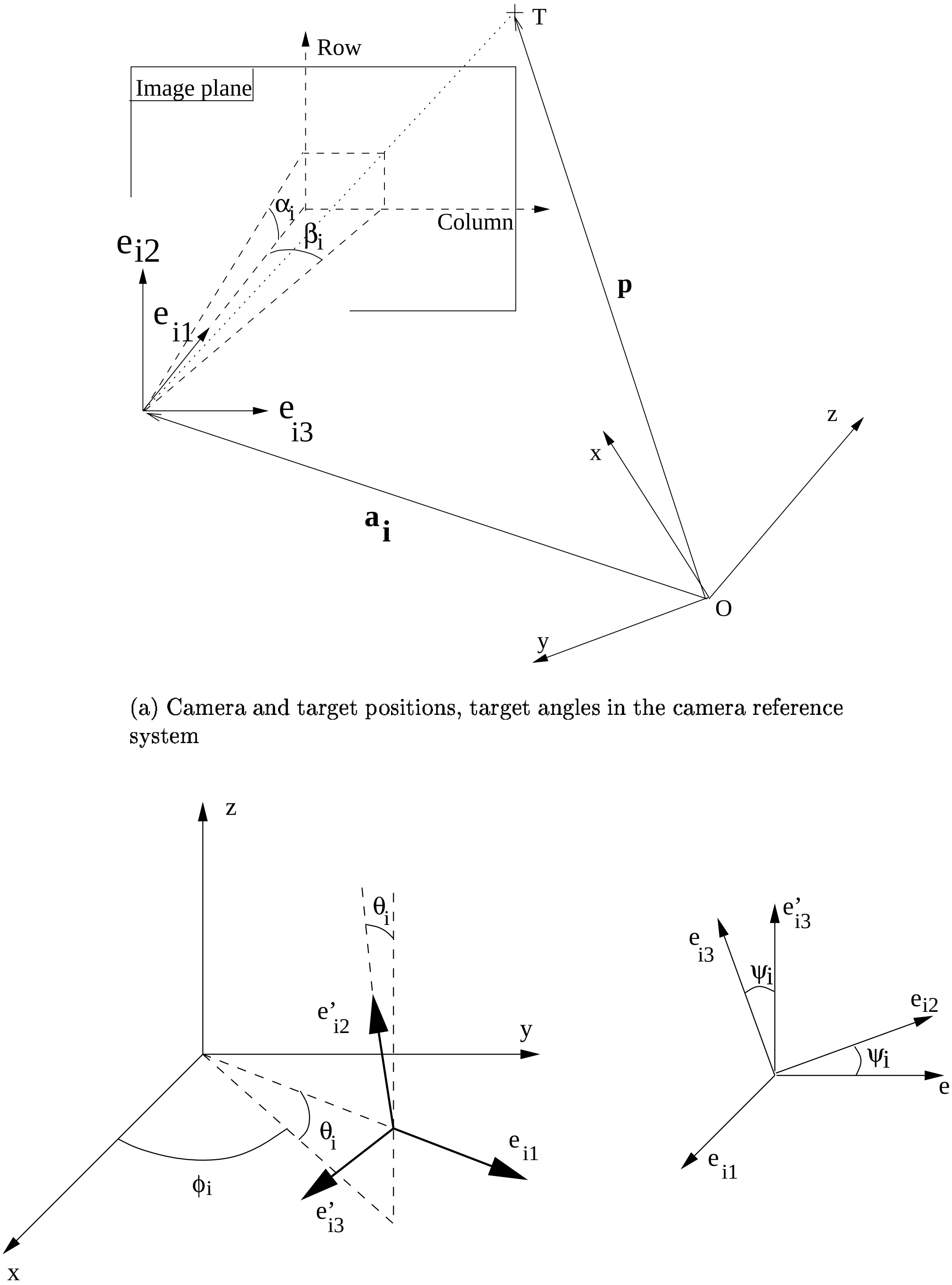
The video system typically comprises two cameras viewing a measurement volume from different directions. Figure 4.1 summarises the system geometry and variables for a single camera.

The vectors are expressed in a coordinate system (O,x,y,z) chosen for experimental convenience. The target T is at a vector position $\mathbf{p}$, and each camera is at $\mathbf{a}_i$ (i denotes the camera number). A single camera has three orthogonal unit vectors which define the viewing axis ($\mathbf{e}_{i1}$), the inclination direction ($\mathbf{e}_{i2}$) and the azimuth direction ($\mathbf{e}_{i3}$). The camera measures the angles of inclination (α_i) and azimuth (β_i) of the target in its own coordinate system. At this stage the model assumes that the camera image coordinates (row, column) can be translated into angles of inclination and azimuth. This is the video image calibration described in section 4.7.

The unit axes for each camera can be expressed in terms of three angles relative to the fixed reference system (O,x,y,z). These angles are the inclination (θ_i) and azimuth (ϕ_i) of the viewing axis $\mathbf{e}_{i1}$, and the angle (ψ_i) about the viewing axis of the camera's azimuth direction from the plane containing the view axis and the vertical (see Figure 4.1(b)).

4.2.1 Basic model

In terms of the angles defined above, the unit vectors for camera i are,



(a) Camera and target positions, target angles in the camera reference system

(b) Camera angles in the fixed reference system

Figure 4.1: Video system: geometry and notation

$$\mathbf{e}_{i1} = (\cos\theta_i \cos\phi_i, \cos\theta_i \sin\phi_i,\ \sin\theta_i) \tag{4.1}$$

$$\begin{aligned}\mathbf{e}_{i2} &= (-\sin\theta_i\cos\phi_i\cos\psi_i+\sin\phi_i\sin\psi_i,\\ &\quad -\sin\theta_i\sin\phi_i\cos\psi_i-\cos\phi_i\sin\psi_i, \cos\theta_i\cos\psi_i) \qquad (4.2)\\ \mathbf{e}_{i3} &= (\sin\theta_i\cos\phi_i\sin\psi_i+\sin\phi_i\cos\psi_i,\\ &\quad \sin\theta_i\sin\phi_i\sin\psi_i-\cos\phi_i\cos\psi_i, -\cos\theta_i\sin\psi_i) \qquad (4.3)\end{aligned}$$

Using these camera axes, the inclination (α_i) and azimuth (β_i) of the target as measured by camera i are,

$$\begin{aligned}\tan\alpha_i &= \frac{(\mathbf{p}-\mathbf{a}_i).\mathbf{e}_{i2}}{(\mathbf{p}-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.4)\\ \tan\beta_i &= \frac{(\mathbf{p}-\mathbf{a}_i).\mathbf{e}_{i3}}{(\mathbf{p}-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.5)\end{aligned}$$

Assuming that the position of the camera and its viewing angles are known, the coordinates of $\mathbf{p}$ are the three unknowns. By using a single camera, only two equations can be written and the system is underdetermined. This shows why two cameras at least are needed to solve for the three-dimensional position of a target.

4.2.2 Linearised model

A linearised version of the model is useful as it can be inverted directly to give target position as a linear function of the camera measurements. The assumption made to linearise the equations is that the target position ($\mathbf{p}$) is close to a reference position ($\mathbf{p}_0$) which lies in the measurement volume close to the view axis of all cameras (close is defined relative to the distance from the camera to the target). The target position can be written $\mathbf{p}=\mathbf{p}_0+\mathbf{p}\prime$, and then the system equations become,

$$\begin{aligned}\tan\alpha_i &= \frac{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i2}}{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.6)\\ &= \frac{\mathbf{p}\prime.\mathbf{e}_{i2}}{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i1}}+\frac{(\mathbf{p}_0-\mathbf{a}_i).\mathbf{e}_{i2}}{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.7)\\ \tan\beta_i &= \frac{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i3}}{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.8)\\ &= \frac{\mathbf{p}\prime.\mathbf{e}_{i3}}{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i1}}+\frac{(\mathbf{p}_0-\mathbf{a}_i).\mathbf{e}_{i3}}{(\mathbf{p}_0+\mathbf{p}\prime-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.9)\end{aligned}$$

If the assumption holds then $|\mathbf{p}\prime|$ is much smaller than $|\mathbf{p}_0-\mathbf{a}_i|$ and can be ignored in the denominator. The constants (corresponding to the angular position of the reference point $\mathbf{p}_0$) can be written as offset angles, giving:

$$\Delta\alpha_i = \tan\alpha_i-\tan\alpha_{i0} \simeq \frac{\mathbf{p}\prime.\mathbf{e}_{i2}}{(\mathbf{p}_0-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.10)$$

$$\Delta\beta_i = \tan\beta_i-\tan\beta_{i0} \simeq \frac{\mathbf{p}\prime.\mathbf{e}_{i3}}{(\mathbf{p}_0-\mathbf{a}_i).\mathbf{e}_{i1}} \qquad (4.11)$$

with:

$$\tan \alpha_{i0} = \frac{(\mathbf{p}_0 - \mathbf{a}_i) . \mathbf{e}_{i2}}{(\mathbf{p}_0 - \mathbf{a}_i) . \mathbf{e}_{i1}} \tag{4.12}$$

$$\tan \beta_{i0} = \frac{(\mathbf{p}_0 - \mathbf{a}_i) . \mathbf{e}_{i3}}{(\mathbf{p}_0 - \mathbf{a}_i) . \mathbf{e}_{i1}} \tag{4.13}$$

The point P_0 is a fixed point in the measurement volume. In practice it can be chosen arbitrarily, i.e. the coordinates $(p_{0_x}, p_{0_y}, p_{0_z})$ are arbitrarily fixed. P_0 can be specified in two ways. Its coordinates in the reference system can be user-defined (values for $p_{0_x}, p_{0_y}, p_{0_z}$ are given by the user). There is no need to know the physical position of P_0 nor there is a need to match it with a physical object. The angles α_{i0} and β_{i0} are derived from $\mathbf{p_0}$ with Equations (4.12) and (4.13). The alternative to this solution is to define P_0 by its angles α_{i0} and β_{i0} and derive $\mathbf{p_0}$ from these angles. This solution is implemented in practice by choosing a specific point in the first frame of the two camera recordings (e.g. initial position of the target) and by extracting its inclination and azimuth angles (α_{i_0} and β_{i_0} respectively).

4.2.3 Model uses

The different ways to apply the video model are:

- System calibration. The position and orientation of the cameras is calculated from a set of calibration targets of known position in the fixed reference system.
- Position measurement. Using known camera positions and orientations the model is used to estimate the positions of a target in three dimensions from a set of measurements.
- More general use. The basic equations ((4.4) and (4.5)) could be used for any imaging situation where camera image coordinates (expressed as angles) need to be related to geometrical position of the target relative to the camera and its attitude.

4.3 System calibration

The calibration of the video system is the first use that can be made of the model described in the previous section. In fact it is the necessary first step in the use of the video measurement system, as the position and viewing angles of the two cameras are required before the position of a target can be calculated. What is called system calibration here is the determination of the vectors $\mathbf{a}_\mathrm{i}$ and of the camera orientation angles θ_i, ϕ_i, and ψ_i. Equations (4.4) and (4.5) are used for that purpose, with the known position of points in the camera image (calibration points). Since there are 6 parameters to estimate for each camera (3 coordinates and 3 angles), a minimum of 3 calibration points is required. However, the 3 points define a plane which should not be parallel to the planes defined by any two axes of the cameras reference system. In order to suppress this possible ambiguity, four calibration points were used for the measurements described in this report.

4.3.1 The Levenburg-Marquadt (LM) method for non-linear model inversion

Equations (4.4) and (4.5) are non linear with respect to the unknown parameters and have to be inverted numerically. A standard way to invert a non-linear model is by maximum likelihood fitting of the model to some measured data [56]. For this purpose the Levenburg-Marquadt (LM) method [56] is used. It requires analytical expressions of the model equations and their derivatives with respect to all the model parameters, and a good first guess for the parameters to be estimated.

The standard code which implements the LM algorithms from [56] was already available at Cranfield University. The non-linear equations (4.4) and (4.5) can be used not only for system calibration but also for target position measurement. However, in order to gain processing time, the linearised version of the model is used for position measurement, as will be explained in section 4.4.

4.3.2 Determination of a first guess for the camera parameters

For initialisation the LM algorithm requires first guess values of the unknowns to be determined. For calibration, a good first guess is best estimated using a crude direct measurement of the camera positions and orientation angles in the fixed reference system.

An accurate first guess is necessary to make the algorithm converge faster. But more important is the possibility that the model finds a local minimum which is not the wanted least square solution. In particular such local minima can occur for values which are symmetrical to the desired solution. In practice, erroneous solutions can be found at angles rotated by π compared to the real solution, and for coordinates of $\mathbf{a}_\mathrm{i}$ with signs opposite to the real coordinates. For that reason it is important to look at the true geometry of the imaging system in order to initialise the least square fitting algorithm with values close to reality and obtain the correct solution.

4.3.3 Accuracy of the calibration

The uncertainty on the position and orientation angles of the cameras is determined by the angular accuracy at which the cameras are able to retrieve the position of a given point in their field of view. Since the camera parameters are calculated from the specification of the angles α_i and β_i, it is logical to expect that the accuracy of the calibration is related to the accuracy at which these angles are specified. Although the LM algorithm provides a numerical estimate of the errors on the retrieved parameters [56], it is useful to have an insight into the aspects which influence the accuracy of the system calibration. In this section the radial and longitudinal accuracies are expressed. They correspond respectively to the position accuracies perpendicular to and along the calibration targets. Since they are defined in two perpendicular directions, the figures and expressions below are derived in plane geometry, where two calibration targets are sufficient to estimate the accuracies.

Radial accuracy

The radial accuracy is the accuracy of the camera position perpendicular to the calibration targets. In Figure 4.2, the two calibration targets are separated by a baseline b, and the camera C is at a distance x from the line defined by the calibration points. The viewing angle between the two calibration points at the camera is 2θ.

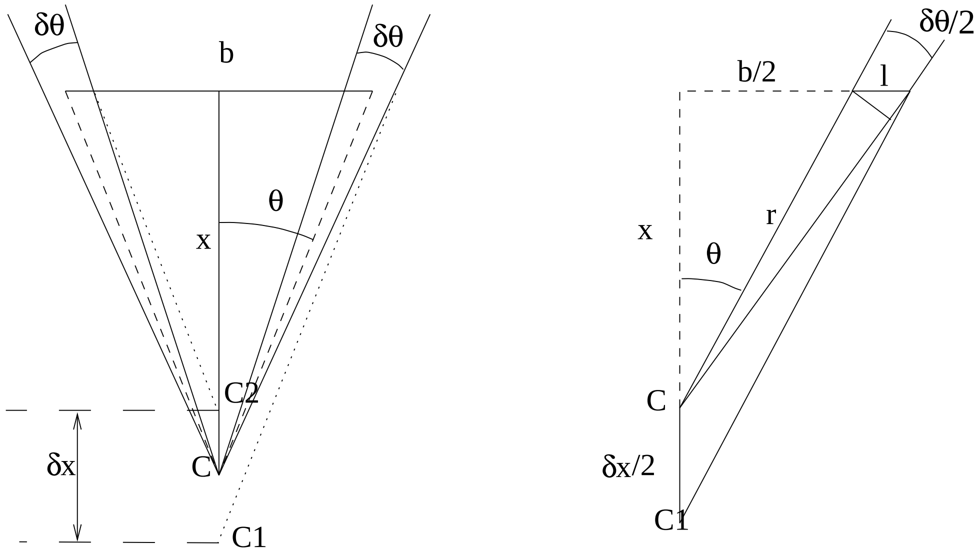


Figure 4.2: Geometry for the determination of the radial accuracy

The angular position of the calibration points is determined with an accuracy $\delta\theta$. The uncertainty $\delta\theta$ means that position C of the camera can vary between the boundaries C_1 and C_2, which define the radial position accuracy δx. From the right sketch on Figure 4.2, the geometry of the system yields the following relation:

$$\frac{\delta x/2}{x} = \frac{l}{b/2} = \frac{\frac{r\delta\theta}{2\cos\theta}}{b/2} = \frac{r\delta\theta}{b\cos\theta} \tag{4.14}$$

Since $\cos\theta = x/r$, the uncertainty δx on the radial position of the camera can be written:

$$\delta x = \frac{2r^2}{b}\delta\theta \tag{4.15}$$

The camera angular accuracy $\delta\theta$ is fixed for a given camera, and a value will be calculated in section 4.7. From Equation (4.15), it is clear that, at a given range r which is usually determined by the application, the calibration targets should be placed as far apart from each other as possible to increase the value of b, in order to obtain the lowest possible radial uncertainty. The value of b is limited only by the field of view of the camera: obviously the calibration points have to be present in the image to be useful. A numerical application for Equation (4.15) will be given later in this chapter, for the specific case of wheat motion retrieval.

Longitudinal accuracy

The longitudinal accuracy of the camera is the accuracy of its position along the line defined by the calibration targets. Figure 4.3 shows the calibration targets aligned with the camera at position C. Because of the camera angular uncertainty $\delta\theta$, the calibration points A and B could in the worst case be viewed by the camera in the positions A' and B' showed in Figure 4.3, and the estimated camera position would differ from the true position by the amount δy.

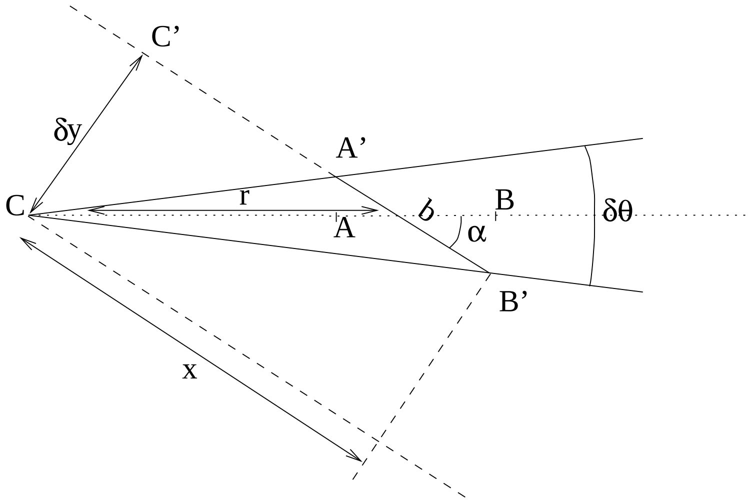


Figure 4.3: Geometry for the determination of the longitudinal accuracy

From Figure 4.3, the following relation can be written, in the small angle approximation for $\delta\theta$:

$$\delta\theta = \frac{b \sin\alpha}{r} = \frac{b\delta y}{r^2} \tag{4.16}$$

So the uncertainty δy on the longitudinal position of the camera is:

$$\delta y = \frac{r^2}{b}\delta\theta \tag{4.17}$$

As for the radial accuracy, Equation (4.17) shows that a better accuracy can be reached by maximising the distance between the calibration points.

4.3.4 Practical implementation of the calibration

In practice the calibration is performed in the following steps:

- The calibration points in the fixed reference system are defined. A frame of reference defines the fixed reference system of the experiment. Points on this frame of reference are marked and define the calibration points.
- The frame of reference is placed in the field of view, and the video camera records it for a few seconds. In practice, only a still image is required, so the camera can be used in a still mode if it has one.

- An image containing the reference frame and the calibration points is extracted from the video camera. The reference frame does not need to be in the field of view of the camera during the remainder of the experiment. It can be removed if it impinges the motion of the target of interest. However the camera should not be moved at any time after the reference frame is removed from the field of view.

- The associated inclination and azimuth angles of the calibration points are determined from the image. In practice the row and column number of the points are translated into angles by an image calibration function (see section 4.7). The inclination and azimuth angles for the calibration points are specified in an input text file, together with the accuracy on these angles, also determined in section 4.7.

- The procedure which implements the LM algorithm requires that the user first loads the input file containing the angle values and accuracies on the calibration points. Then the LM algorithm itself calculates the camera parameters, after the user has specified the values of the input file to use for model fitting, the parameters to be varied in the model (in the general case, all of them), and a first guess for the model parameters.

4.4 Target position measurement

From now on the derivation will assume that two cameras are used for the measurements, so the subscript i introduced in section 4.2 can take the values 1 and 2. The measurement of the position of a target is the calculation of the coordinates of $\mathbf{p}'$ (see section 4.2). The coordinates of a target are calculated with respect to a fixed point P_0 in the measurement volume, so the linearised model equations can be applied.

4.4.1 Equation system to solve for $\mathbf{p}'$

The coordinates of $\mathbf{p}'$ are calculated with the set of Equations (4.10) and (4.11), in which i takes the values 1 and 2:

$$\mathbf{p}'.\frac{\mathbf{e_{12}}}{(\mathbf{p_0}-\mathbf{a_1}).\mathbf{e_{11}}} = \Delta\alpha_1 \tag{4.18}$$

$$\mathbf{p}'.\frac{\mathbf{e_{22}}}{(\mathbf{p_0}-\mathbf{a_2}).\mathbf{e_{21}}} = \Delta\alpha_2 \tag{4.19}$$

$$\mathbf{p}'.\frac{\mathbf{e_{13}}}{(\mathbf{p_0}-\mathbf{a_1}).\mathbf{e_{11}}} = \Delta\beta_1 \tag{4.20}$$

$$\mathbf{p}'.\frac{\mathbf{e_{23}}}{(\mathbf{p_0}-\mathbf{a_2}).\mathbf{e_{21}}} = \Delta\beta_2 \tag{4.21}$$

with

$$\begin{aligned}\Delta\alpha_i &= \tan\alpha_i - \tan\alpha_{i0}\\ \Delta\beta_i &= \tan\beta_i - \tan\beta_{i0}\end{aligned} \tag{4.22}$$

This set of equations can be developed into a system of 4 equations and 3 unknowns:

$$\mathbf{E}.\mathbf{p}' = \mathbf{c} \tag{4.23}$$

with

$$\mathbf{E} = \begin{bmatrix} \frac{e12_x}{s_1} & \frac{e12_y}{s_1} & \frac{e12_z}{s_1} \\ \frac{e22_x}{s_2} & \frac{e22_y}{s_2} & \frac{e22_z}{s_2} \\ \frac{e13_x}{s_1} & \frac{e13_y}{s_1} & \frac{e13_z}{s_1} \\ \frac{e23_x}{s_2} & \frac{e23_y}{s_2} & \frac{e23_z}{s_2} \end{bmatrix} \tag{4.24}$$

$$\begin{array}{l} s_1 = (\mathbf{p_0} - \mathbf{a_1})\,.\mathbf{e_{11}} = (p_{0_x} - a_{1_x})e_{11_x} + (p_{0_y} - a_{1_y})e_{11_y} + (p_{0_z} - a_{1_z})e_{11_z} \\ s_2 = (\mathbf{p_0} - \mathbf{a_2})\,.\mathbf{e_{21}} = (p_{0_x} - a_{2_x})e_{21_x} + (p_{0_y} - a_{2_y})e_{21_y} + (p_{0_z} - a_{2_z})e_{21_z} \end{array} \tag{4.25}$$

$$\mathbf{p}' = \begin{pmatrix} p'_x \\ p'_y \\ p'_z \end{pmatrix} \tag{4.26}$$

$$\mathbf{c} = \begin{pmatrix} \Delta\alpha_1 \\ \Delta\alpha_2 \\ \Delta\beta_1 \\ \Delta\beta_2 \end{pmatrix} \tag{4.27}$$

4.4.2 Numerical solution for $\mathbf{p}'$

Equation (4.23) is solved using the Singular Value Decomposition (SVD) of **E**. Calculating the SVD of a matrix is a powerful and easy way to find a least square solution of overdetermined linear systems of equations. Details on the theoretical basis of SVD is given in Appendix B, as it forms the basis of the numerical solving of Equation (4.23). There are other techniques applicable for solving systems of linear equations but the SVD technique is preferred here, as it only requires a matrix formulation of the equation system, and can be programmed in the IDL language using its capability to directly compute matrix operations. The procedure for numerical solving of Equation (4.23) is:

1. The positions (row and column number) of the target are read from a text file which is the output of the tracking algorithm to be detailed in section 4.6. The file contains as many lines as there are frames in the video sequence, and each line is composed of 3 columns, namely the frame number, the column number and the row number. There are two files, one per camera, which are both read in the IDL program.

2. The row and column number for each frame and both cameras are translated into angular coordinates α_i and β_i, using the image calibration functions derived in section 4.7.

3. The coordinates of the point P_0 are entered by the user, either in true coordinates in the fixed reference system, or row and column numbers which are translated into the angular values α_{i0} and β_{i0}.

4. The positions and orientation angles of the two cameras, obtained from the calibration procedure of section 4.3, are read from two text files.

5. From the position and orientation angles of the cameras, the unit vectors defining the coordinate systems of the two cameras are calculated using Equations (4.1) to (4.3).

6. The matrix **E** and vector **c** are expressed (Equations (4.24) and (4.27) respectively).

7. The SVD of **E** is calculated by the 'svdc' procedure included in the IDL language. The procedure outputs the matrices **U**, **W**, and **V** presented in Appendix B.

8. Small values in **W** (inferior to a threshold $\epsilon = 10^{-5}$) are set to 0 (see Appendix B for a justification).

9. The backsubstitution, using the 'svsol' procedure in IDL, returns the coordinates of $\mathbf{p}'$ for each frame of the data video sequence.

The processing time of the procedure above is negligibly small as the matrix operations on **E**, $\mathbf{p}'$, and **c** are computed directly in IDL, without the need to introduce loops in the code. The starting point of the procedure is to read in a text file the positions of the target in the two camera video sequences. The two following sections will now present how these positions are obtained from the video data.

4.5 Video data management

The determination of the positions of a given target in the video sequences requires some processing on the video files. For that reason it is necessary to first input the video data recorded by the cameras into a PC. This section describes how the video data recorded by the cameras is translated into a workable format for the object tracking to be performed. Section 4.6 will then present the tracking algorithm of an object in the video files.

4.5.1 System overview

The two video camcorders used for the measurements are commercially available from most video retailers at the time of this report (2000). They are digital camcorders SONY DCR-TR7000E using the Digital 8 format. They contain a digital output channel (IEEE 1394 serial connection) which is linked to a personal computer via a 1394 FireWire interface card from Digital Origin Inc. The interface card is delivered with the MotoDV^{TM} software which transfers DV (Digital Video) clips into the computer and saves them as QuickTime movies which can be imported into QuickTime-compatible applications such as Adobe Premiere^{TM} for video editing.

The personal computer used for image processing and target position retrieval works with a Celeron processor running at 350 MHz with 128 Mbytes of RAM and a graphics card. For reasons explained in section 4.6, the AVI (Audio Video

Interleave) film format is used by the target tracking algorithms . The Adobe Premiere^{TM} software is used to convert the default QuickTime format delivered by MotoDV into AVI format and to select or edit sequences of the films if necessary.

The processing does not need to be executed in real time, so the film recording can be performed initially without the use of the PC. The system is therefore easily transportable and well suited for outdoors experimentation.

4.5.2 System hardware

The video cameras

The SONY DCR-TR7000E digital video cameras are standard cameras commonly used for mass consumption. The rapid increase of the market for this kind of product is likely to result in a decrease of the price of these items. To give an idea of the decrease to expect, it is interesting to note that the first camera was bought in August 1999 for £700 and the second in April 2000 for £400.

The cameras perform the standard recording/playing functions required for the project, and a wealth of other functions which were not used for the research. The Digital 8 format is a digital video format which uses Hi-8 tapes as a memory support. Old analogue films on Hi-8 tapes can be inserted into the camera for digitisation. The Digital 8 format is currently the competitor of the DV format, which simply uses a memory card as a support. The batteries on the cameras (Lithium Ion) give to the cameras an autonomy of 1h30 to 2h00 of constant use (i.e. recording or playing). Such an autonomy is ideal for field use. The cameras have a DV output socket where a IEEE 1394 serial connection can be plugged.

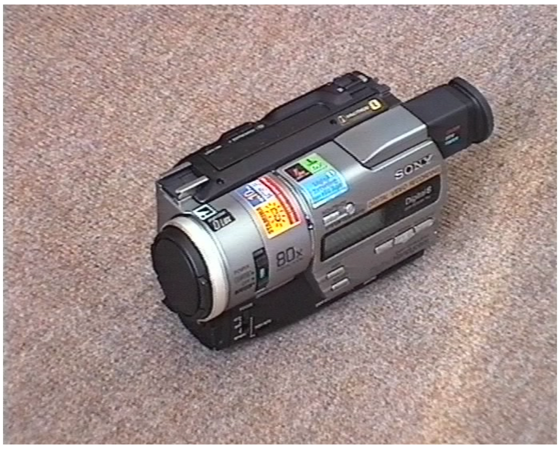


Figure 4.4: The SONY DCR-TR7000E digital video camera

The IEEE 1394 connection, the FireWire interface card, and the computer

The IEEE 1394 connection relates the video cameras to a 1394 FireWire card, which provides the necessary interface between the video data on the Hi-8 tape and the computer. The specifications of the computer (Celeron 350 MHz, 128 Mbytes RAM, 3D Rage Pro AGP 2x graphics card) are by no means demanding by current standards. It will however prove to be sufficient for the purposes of the research. It should be noted however that, in the future, the cost of computers is bound to decrease, as their performances are bound to increase. In such a context, the performance of the system in terms of processing time is likely to augment as its cost will come down. This concerns not only the computer but also the interface card.

4.5.3 System software

MotoDV

MotoDV is the software available with the FireWire interface card. It provides an interface with the user to drive the camera and to capture video sequences from the Digital 8 tapes to a QuickTime format movie file. The MotoDV software is used to select the video sequence of interest via the classic camera functions (Play, Fast Forward, etc...). The video capture can be performed either for all frames recorded or in the 'time lapse' mode, where the user specifies the number of frames to capture in a given time interval and the total duration of the capture. The time lapse mode is useful for applications where the full frame rate of 25 frames/s is not necessary. The format of the output file is QuickTime (file extension '.mov').

Adobe Premiere^{TM}

Adobe Premiere^{TM} is a separate software useful for video editing. It features many functions which are not relevant to the research here, but useful for conventional video editing users (text addition, film transitions, etc...). The software is used in the research for two reasons. First, it is difficult with MotoDV to select the start and end frames of a film sequence with exact accuracy. This is easily done with Adobe Premiere^{TM} as a video sequence can be truncated at the exact frame required. This aspect is particularly important for synchronisation with another camera, as it is important in this case to know exactly which frame of the first camera corresponds to which frame of the second camera. Synchronisation of the two cameras is detailed in section 4.8. The second use made of Adobe Premiere^{TM} in the research is to convert the default QuickTime format given by MotoDV into the AVI format.

4.5.4 Video data acquisition procedure

Once the motion of the object of interest is recorded, the camera is connected to the computer via the IEEE 1394 serial connection and the FireWire interface card. The sequence of interest in the film is captured with MotoDV. The capture here does not need to be accurate to the frame, since Adobe Premiere can be used to select exactly the sequence of interest. MotoDV imposes a limitation of 2 Gbytes for the total size of the output QuickTime file, which corresponds to a film duration of about 10 minutes if all 25 frames/s are captured. If a longer time span is required, then the user can either switch to time lapse mode to capture less than 25 frames/s, or capture several real time video sequences in separate files.

Once the QuickTime video file is saved, Adobe Premiere is used to select the exact part of the total video sequence which is to be used for object tracking. Selecting a relevant part of the film is left to the judgement of the user, as it depends on scientific interest and significant time length for the application. For object tracking, the required video file format is uncompressed AVI. Adobe Premiere^{TM} allows to export a selected part of the film sequence in this format. The export options should include no compression scheme, and millions of colours (1 Byte per colour channel per pixel). The frame rate can also be changed.

Such a video format leads to very large file sizes: an uncompressed AVI file, with 3 bytes per pixel and 25 frames/s, corresponding to one minute of real time, requires approximately 1.8 Gbytes of memory. From the experience acquired during the research, it is recommended to keep a minimum amount of uncompressed AVI files stored in the computer, especially if several video files are to be processed. The best scheme is to convert the video sequence to AVI, perform the necessary processing on it, and delete the AVI file after use. The most suitable support for storage of the video data remains the original Digital 8 tape.

4.5.5 Conclusion

The video data is easily converted into the necessary AVI format for processing, using the commercially available technology. Digital video camera still experience at the moment significant technology advances, and it is difficult to foresee if the method described in this section will still be applicable in future years. In particular, the use of Digital 8 tapes as a support for digital video data is justified more by 'historical' reasons linking the Digital 8 standard to the older analogue Hi-8 format. It is likely that future video cameras will move on to use different standards. However this does not mean that the data transfer into the computer will not be possible as described in the above sections. It is probable that the MotoDV and Adobe Premiere software will have improved capabilities in future versions, but the transfer into the uncompressed AVI format should still be a feature of these future releases. The AVI format is a very common video format which is likely to be used in the future.

In conclusion, there is at the moment no reason to suppose that the video data acquisition described in this section will not be possible with forthcoming hardware and software. The system and methods may have to be adapted to cope with future changes but will still be relevant. The changes to come point towards an amelioration of the system, as both hardware and software will become more capable and cheaper.

4.6 Video data processing

Section 4.4 showed how the coordinates of an object are retrieved from the knowledge of the position of this object in each frame of the image on both cameras. The position of the object in a frame is given by its column and row number, and it is the purpose of this section to explain how these column and row numbers are retrieved from the image. Since the object of interest is moving it is necessary to track its position for each frame. Section 4.5 described how the video film is imported into a computer and converted to the AVI format. This section justifies the choice of the AVI format for processing, and details the strategy employed for the recognition of a particular shape in the image. The tracking algorithm based on this strategy is described, tested, and its potential further developments are discussed in the concluding lines of the section.

4.6.1 The AVI format

The AVI data format (extension '.avi') is one of the most popular video formats developed by Microsoft. This format was developed to play videos in the Windows environment. There is detailed information available about the structure of AVI files and about their editing and conversion possibilities in [62]. It is briefly summarised here.

An AVI file is composed of a file header containing information on the audio and video format (number of audio streams, video image size, number of frames, etc...), followed by a list of audio and video data (the 'movi' list) . This list contains as many elements as there are frames in the video file. Each element of the list is composed of the audio and video data for a single frame. In the specific case of uncompressed AVI, the video data for a single frame is stored as a bitmap image. In a bitmap image, the data are interleaved by pixel, and each pixel is described by 3 values defining the red, green, and blue colour bands. Each colour band being specified by a 1 byte unsigned integer, it takes a value ranging from 0 to 255.

Uncompressed AVI allows the bitmap data to be regularly spaced in the file. If a compression scheme is used, the size of each element of the 'movi' list will differ, which makes the retrieval of the pixel information of each frame much more complex. An exact knowledge of the compression technique is required in this case. In addition to this, the processing to be done on each frame requires that the value of the three colour bands of each pixel is known. It is for this reason that the target tracking algorithm deals only with uncompressed AVI files. For the same reasons there is no audio information kept in the processed AVI files. The source code (programmed in C) necessary to manipulate AVI files and extract the pixel values for each frame was available at Cranfield University. The software described in the following sections builds on this available code.

4.6.2 Background on object recognition and design rationale

Object recognition in digital image processing is a very difficult problem to solve in the general case and is still the subject of active research [60]. The two main areas which pose difficulties in object recognition concern (1) the detection of the edges of a given object in contrast with the background, and (2) the matching of the shape extracted by the edge detection algorithms to known 'ideal' features for object recognition. In the general case, these two aspects require some solid processing techniques.

Rather than trying to find solutions for the entire issue of object recognition, it is more useful to use the experience of the specialists in the area to find a viable solution to the problem of object tracking of prime concern for the research here. From the first problem in object recognition, the detection of edges, the obvious conclusion is that the form to be extracted in the image should have a high contrast with the background for easier detection. This is not always the case in general, so it is necessary to increase this contrast in the image. For example, in the case of prime interest here (the measurement of crop motion), finding a particular green crop leaf against a background dominated by green elements is virtually impossible. For that reason the element whose motion is to be measured should be marked with a colour contrasting strongly with the background.

The second problem in object recognition, the matching of the extracted shape with a known object, is difficult to solve in the general case, but not if the shape of the feature to track is known *a priori*. Algorithms can be written to find a known shape in the image. The combination of the specificity of the shape to track with its highly contrasting colour make the object to track unique in the image, and therefore easier to find.

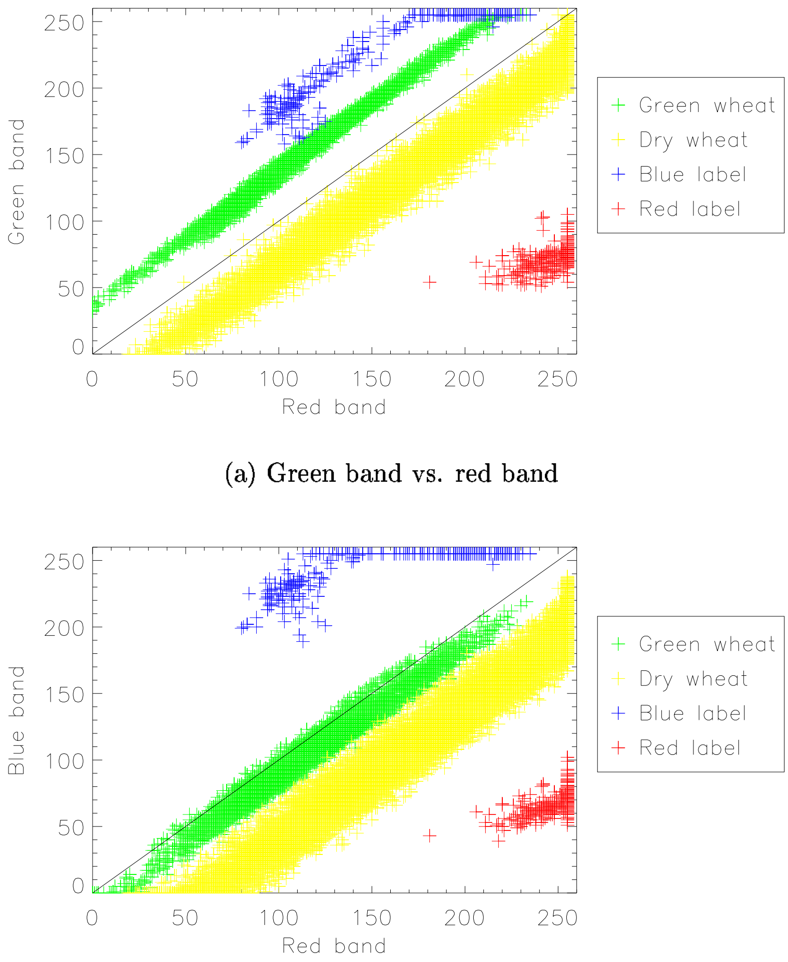
For that reason, the element to follow in the video sequence should be marked with labels of specific colour and shape. In the case of wheat, the labels are chosen to be coloured discs. A round shape is specific in a wheat field background, and remains fairly constant even for slight changes of the viewing angle. These coloured discs are physically glued on the wheat elements to measure.

4.6.3 Quantitative analysis on the choice of the colour of the discs

The discs on the wheat should give a high contrast with the background. The notion of contrast is usually intuitive to the human eye, but it is necessary to justify quantitatively the choice of the appropriate colour. Of the candidate colours for the labels, the most appropriate seem to be red or blue. Other colours resulting from the combination of the 3 basic colour bands do not always provide much contrast against a vegetation background, or they are not as commonly available.

In order to justify the choice of a colour for the discs, an image of a wheat field with red and blue labels glued to a wheat head was taken with one of the video cameras. Figure 4.5(a) shows the colour values in the green band plotted against those in the red band, obtained from pixels in the background (the wheat), and from the red and blue labels separately. Two wheat backgrounds are considered: dry wheat (yellow dominant) and young wheat (green dominant). Figure 4.5(b) is equivalent to 4.5(a), with the blue band on the y-axis instead of the green band.

There are more points on the graphs for the wheat backgrounds since they cover a larger area than the coloured labels in the image. In the two cases of dry and green wheat, the pixels are correlated along a line parallel to the grey scale line (the line joining the [0,0] point to the [255,255] point). This means that the background has a uniform colour. The more spread the points are along the regression line, the less uniform the background is. So the green wheat background is more uniform in colour than the dry wheat background, since the yellow points on the graphs are more spread than the green points. Figure 4.5(a) shows that the green wheat background pixels are shifted towards higher values in the green band, compared to the dry wheat background pixels, which obviously do not look as green. Comparison of Figures 4.5(a) and 4.5(b) shows that the dry wheat background contains pixels



(a) Green band vs. red band

(b) Blue band vs. red band

Figure 4.5: Contrast analysis on the coloured labels against the background wheat

values which are higher in the red and green bands than in the blue. The resulting colour therefore tends towards yellow (the colour of dry wheat). The green and dry wheat backgrounds plotted in Figure 4.5 represent the two extremes in the colours to expect throughout the season. The variation of the background colour from a wheat field varies in the season from a green-shifted uniform background to a yellow-shifted background of a less uniform colour. In this context it is possible to compare the pixels from the coloured labels to these backgrounds in order to get a better idea of which label gives the best contrast.

As expected, the red label pixels have higher values in the red band and similarly for the blue label pixels in the blue band. Two interesting features are noticeable from the plots. First, the red label pixels are more compactly grouped in a specific region of the 3D colour space than the blue label pixels. Visually, this means that the blue label in the image has a less uniform colour. Some pixels are even saturated in blue (they take the value 255 in the blue band) and in green, and have simultaneously high red band values. In other words, some pixels on the blue labels are very bright, probably due to direct reflection of the sunlight in the direction of the camera. This effect in not present on the red label, whose pixels do sometimes saturate in the red band, but not in the others. For that reason, it seems that the red label is more appropriate as it give a more constant and well-defined colour. In addition to this, Figure 4.5(a) shows that the region occupied by the blue label pixels in the red-green colour subspace is in contact with the region of the green wheat background pixels. This is not the case for the red label pixels. In other words the red label gives a higher contrast to the wheat background than the blue label. With the dry wheat background this feature is less pronounced, but some of the bright pixels of the blue label closely approach the dry wheat region.

The conclusion from the analysis is that red labels are more adequate for tracking with a background composed of vegetation. The more quantitative study above confirms the fact that the red labels were visually contrasting more than the blue one, and justifies the choice of the red colour. Other colours than red and blue have not been presented here but it is likely that any colour which is a combination of red, green and blue will have pixels lying in the centre part of the 3D colour space, where the wheat background also lies. For that reason primary colours are best suited to provide a higher contrast.

The decomposition of an image into its three colour bands as shown in Figure 4.5 can be repeated in other applications, for different backgrounds. It is a general methodology which allows to determine the most suitable colour of the target to track in the image. There exists numerical methods to quantify the contrast from colour decompositions, but they have not been used here as a simple look at the plots of Figure 4.5 is sufficient to choose the appropriate colour.

4.6.4 Methods for object matching

It is now clear that the shape to find in the image is that of a red disc. The size of the disc in the image depends on its physical size and on the distance between the camera and the disc. Typically, for a disc of 15 mm diameter and with the camera at about 2 m from the target, the red discs cover on the image about 30 pixels along their diameter. Two methods have been implemented to find the red discs in each frame: the thresholding method and the correlation method.

The thresholding method makes use of the colour information in the image. The colour to find in the image is known, fairly constant across the disc, and highly contrasting with the background: it is set in the thresholding algorithm as a colour to match (in the three colour bands) within a user-defined threshold depending on the variability of the colour on the label. The thresholding algorithm sets to 0 (black) all pixels which do not have the same colour as the colour to match within the range of the threshold value. As some isolated pixels in the image may have been selected for their similarity in colour with the colour to match, an averaging window is passed across the image and acts as a low pass filter. For each pixel in the image, the average value of all pixels surrounding the centre pixel is calculated and attributed to the centre pixel. The number of pixels used for averaging in the row and column directions is user defined. After averaging (several times if necessary), the remaining non-black pixels in the image correspond to the label to find. The row and column number of its centre are determined and saved in a text file for each frame.

The correlation method uses both the colour and the shape of the label to find. A mask is defined with, at its centre, pixels of the same colour and forming the same shape as the label to find. At the edges of the mask, black pixels are added and form a square shape. Figure 4.6 represent the mask used in the correlation method.

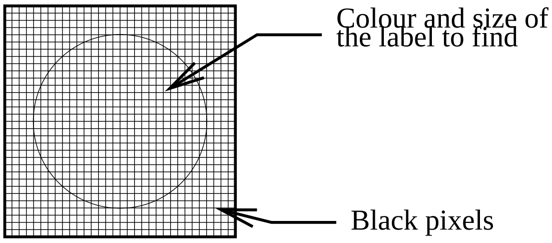


Figure 4.6: Mask used in the correlation method

For a given pixel in the image, the algorithm calculates the correlation coefficient between the pixels of the mask and the pixels surrounding the centre pixel. The number of pixels taken in the image is equal to the number of pixels of the mask. Each pixel being represented by three colour bands, the correlation coefficient is calculated for three-dimensional data points. The mathematical definition of the correlation coefficient in this case in given in Appendix C. As the mask scans the image, the correlation coefficient for each pixel is calculated, and the position of the pixel with the highest coefficient corresponds to the centre of the label to find.

An improvement of the correlation method consists of defining the mask from the image itself, rather than with a shape and some colour values for each of his pixels in the source code of the algorithm. By defining the mask from the first frame of the image, the true shape of the target and its exact colours are used rather than the disc showed in Figure 4.6. It allows slight changes in the shape and colour of the target to be integrated more easily in the correlation algorithm. The next section present the "target selector", which performs this function.

4.6.5 The target selector

The target selector is a Windows interface which allows to select a rectangular area in an image and to store the pixels inside that area in a file. The file can later be read by the target tracking algorithm (see section 4.6.6) and its values are used to define the mask which is scanned across the image for calculation of the correlation coefficients.

The target selector is programmed in Borland C++ Builder, which provides a convenient environment for Windows programming and for the development of Windows applications. A schematics of the program is shown in Figure 4.7. The software is composed of a main window, opened when the application is launched, and a "slave" window (called a "child" window in the Borland C++ builder terminology) which appears when a picture is opened.

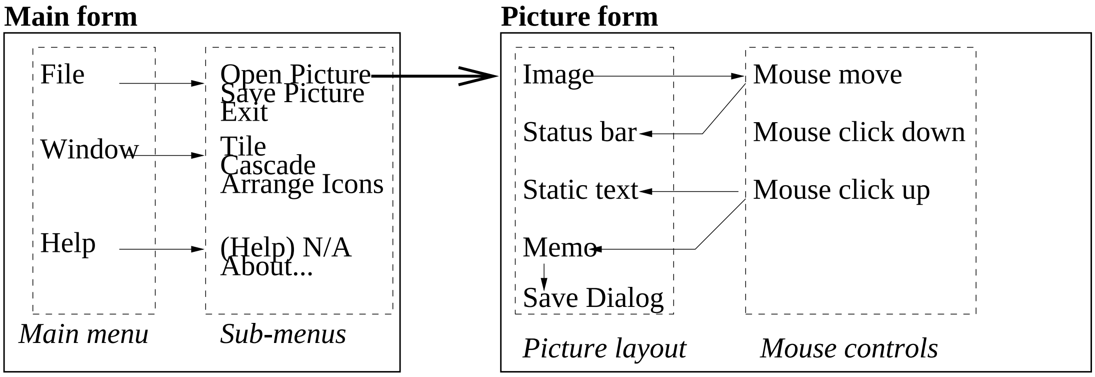


Figure 4.7: Schematics of the target selector program

The main window

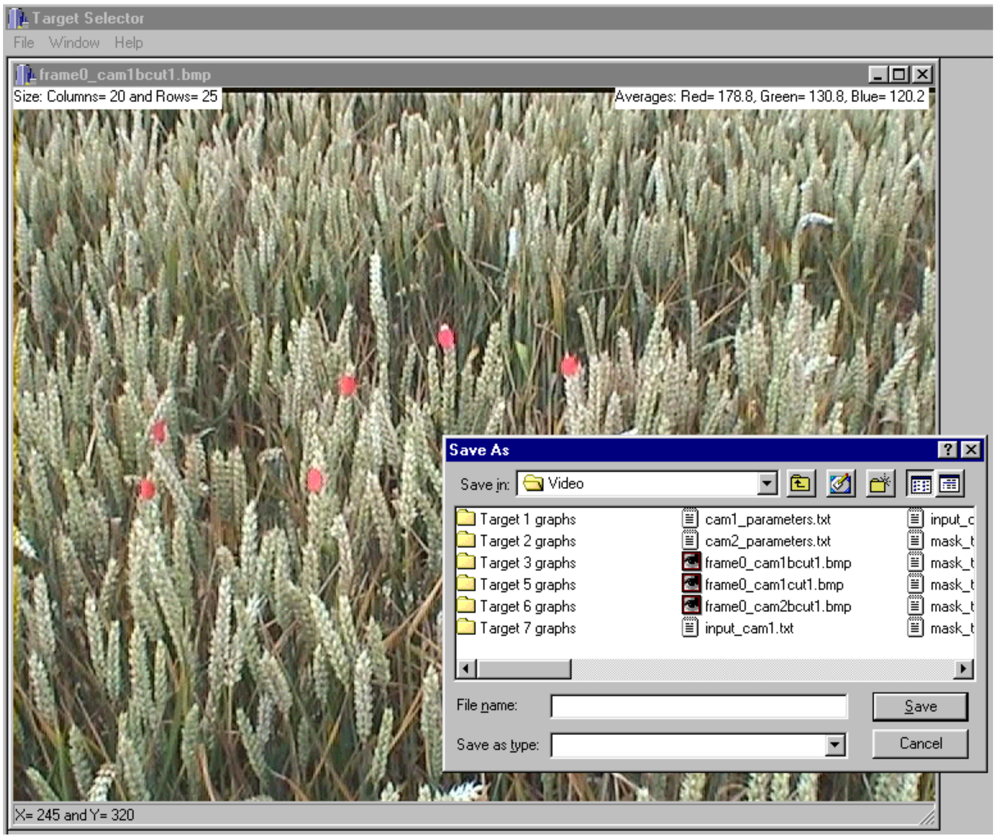
The main window contains a main menu, itself divided into sub-menus. The 'File' sub-menu contains an 'Open' function which produces a dialogue box to open a file from any directory in the computer hard drive. When the 'Open' function is used, the child window containing the image of interest appears. The 'File' sub-menu also contains a 'Save' function which can be used to save changes made to the image. In the current version of the target selector it is not used as no changes can be made to the image file. The 'Save' function is included in view of future potential improvements of the program. The 'Exit' function in the 'File' sub-menu exits the program.

The sub-menu 'Window' contains classic function to manage several windows in the program (tile, cascade, arrange icons), and the 'Help' sub-menu opens the possibility to create help files in future versions of the program, and also gives the current program version.

The picture window

Once the image to open is selected, the program opens a separate window containing the image. As soon as the image is opened, the program constantly checks on mouse events. If the mouse is moved inside the window, the 'mouse move' event calls a function which writes the position of the cursor in the status bar below the image (bottom left corner). When the mouse is clicked down, a rectangle is drawn on the image, with the initial and current position of the cursor as its diagonally opposite corners. The rectangular box allows to visualise on the image the selected area. When the desired area is selected the user releases the mouse and the 'mouse click up' event is called. On this event, two functions are performed. First, the program displays in the top right corner of the image, the average colour of the selected area, in the red, green, and blue channels. The values of the three colour bands in the image are stored in hexadecimal form, so they are first converted into 1 byte integers (range 0-255) before being displayed. The size of the selected window in the image is also shown, in the top left corner. The second function activated by the 'mouse click up' event writes the pixel values (red, green, blue) of all pixels in the selected area to a text window (the 'memo' object) which is not displayed on the screen. The function then opens a Save dialogue box, from which the pixels values in the text window are saved to a text file. It contains fives columns, namely the row number, the column number, and the corresponding red, green and blue values. It is this text file which is opened by the target tracking algorithm to create the mask used for correlation. The first line of the file contains the value of the centre position of the rectangular area.

Figure 4.8 shows the target selector windows, with an image of a wheat field and with the Save dialogue box opened.



Size: Columns= 20 and Rows= 25

Averages: Red= 178.8, Green= 130.8, Blue= 120.2

X= 245 and Y= 320

Figure 4.8: The target selector window used for the definition of the correlation mask

4.6.6 The target tracking algorithms

The target tracking algorithms implement the two methods described in section 4.6.4 to find the position of the centre of the target in each frame of the video file.

Program requirements

There are different types of requirements the tracking needs to meet in order to be useful for the purposes of the research. First the position of the centre of the target needs to be retrieved accurately. The accuracy requirements on the position of the target were set in section 3.6, where it was stated that an accuracy of 1.3 mm is required in slant range direction, 3.3 mm for a displacement in a horizontal plane and a radar incidence angle of 23°. In addition to this, the video data needs to be processed in a short time in order to meet the high time repeatability design driver justified in section 4.1. The performance of the target tracking program will therefore also be assessed on its speed.

The final performance parameter on which the tracking will be evaluated relates to the behaviour of the program in difficult situations where, for example, the speed of the target is such that it appears to be blurred in a frame, or when the camera temporarily loses view of the target due to obstruction by another object in front.

Description of the tracking algorithm, thresholding method

The procedure which calls the thresholding algorithm is chosen from a main menu under an MS-DOS window. From the prompt, the user enters the AVI filename to process and the output filename which will contain the values of the centre position of the label (text file). After the file name is entered, the procedure performs the following operations:

1. The AVI file is opened and the number of frames in the file is obtained from its header. The program then enters a loop in which each frame of the file is processed individually. Steps 2 to 7 describe the functions performed inside this loop.

2. For the frame being processed, the pixels values (three colour bands) are stored in a matrix. The size of the matrix is the product of the number of rows in the frame by the number of columns, times 3.

3. The pixels of the matrix are set to 0 if they are outside the range defined by the colour to match plus or minus the colour threshold. This is the thresholding operation itself, as described in section 4.6.4. In its current version, the colour to match and the colour threshold are specified in the code itself.

4. The resulting values after thresholding are filtered by an averaging filter (see section 4.6.4), and the thresholding loop (step 3) is run again. Step 4 is run as many times as required, until all noisy pixels have disappeared. The number of iterations is left to the judgement of the user, and depends on the nature of the video data. For the specific case of tracking of wheat elements, only one averaging/thresholding loop was necessary.

5. The centre of the remaining non-black pixels is calculated by finding the mean row and column numbers of the non-black pixels.

6. The centre of the remaining pixels is written to the output text file.

7. A green cross is drawn on the frame, at the row and column defining the centre of the label. The aim of this step is to allow the user to control visually that the centre position calculated by the algorithm is accurately retrieved.

8. When all frames have been processed through steps 2 to 7, the program exits.

In its current version, the colour to match, the colour threshold, and the size of the averaging filter are directly specified in the code itself, and not asked at the MS-DOS prompt each time the program is run. Adding this facility is one of the potential improvements of the program. It was not programmed in the original version since the colour to match in the case of the wheat motion measurement was constant in all cases (red).

Description of the tracking algorithm, correlation method

As for the thresholding method, the user is first asked to enter the filenames containing the AVI file, the output text file in which the label centre positions are to be saved, and the text file containing the pixel values of the mask to use for the correlation (obtained from the target selector). After these filenames are specified, the program operates as follows:

1. The AVI file is opened and the number of frames in the file is obtained from its header.

2. The pixel values of the mask are loaded into a matrix. The centre position within the whole image of the rectangular area defining the boundaries of the mask is also loaded into two integers. The program then enters a loop in which each frame of the file is processed individually. Steps 3 to 8 describe the functions performed inside this loop.

3. For the frame being processed, the pixels values (three colour bands) are stored in a matrix.

4. The centre position of the label in the previous frame is called. If the first frame of the file is processed, then the centre position of the mask in the original image is called instead (read from the first line of the mask text file).

5. The correlation coefficient between the mask and a frame sub-window of equal dimensions is calculated, and the value obtained is assigned to the centre pixel of the sub-window. Only pixels in a sub-area of the total image are treated. The centre pixel of the sub-area is obtained from step 4, and its size is defined in the code itself. The correlation coefficient is calculated from the definition of Appendix C, only for the pixels in the sub-area whose colour matches the centre colour of the correlation mask with a range defined by a colour threshold in the code.

6. The highest correlation coefficient is obtained for the centre pixel of the label to match. It is stored in the output text file.

7. The new centre position of the label is passed to a variable, for use in step 4.

8. A green cross is drawn on the frame, at the row and column defining the centre of the label.

9. When all frames have been processed through steps 2 to 8, the program exits.

Processing only a sub-area of the total image is useful to gain processing time. It is possible to do so because the label to track cannot physically move by more than a certain amount. The size of the sub-area is defined so that it is certain that it contains the new position of the label, given the previous position. The specification the sub-window size depends on the video data and the application. The speed of the target is to consider, as well as its maximum range of motion in the image. In the case of the wheat motion measurements, at a camera-to-target distance of about 2 metres, the (square) sub-area size is set to 100x100 pixels. Working only in a sub-area also allows to track several targets in the same image, provided they are not too close to each other. In addition to treating only a sub-area of the total image, the calculation of the correlation coefficient is performed only on pixels of similar colour to that of the centre pixel of the mask (e.g. red in the case of wheat tracking). This allows to reduce further the processing time on each frame.

4.6.7 Performance of the algorithms

The performance of the thresholding and correlation algorithms is assessed in terms of processing time, position retrieval accuracy, and reliability.

Processing time

The processing time of the two algorithms is tested on an AVI file of one minute (1500 frames), where the subject to track is a red label on a wheat head. For comparison, the computer is used in the same conditions. In their form described above, and with the computer whose specifications are given in section 4.5, it takes 21mn 27s to the thresholding algorithm to process the one minute of video data, and 11mn24s for the correlation algorithm. However, these two figures should not be compared directly without a more careful interpretation of their meaning. The thresholding algorithm takes longer because it scans through the whole image to find the red pixels, whereas the correlation algorithm only works in a limited sub-area. To assess the time performance of the two methods, a better indicator is the processing time per number of pixels processed.

The time taken by the correlation algorithm to process the video data (11mn24s) corresponds to 0.46s/frame. Since the sub-window size contains 10^4 pixels (100x100), the processing rate is $r_c = 0.46s/10^4$ pixels. For the thresholding algorithm, the rate is 0.86s/frame. The entire frame is scanned, and contains 720x576 = 414720 pixels, so the corresponding processing rate is $r_t = 0.21s/10^4$ pixels.

Since $r_c > r_t$, the thresholding algorithm is intrinsically faster than the correlation algorithm. The reason why it is not in absolute values is because the two algorithms do not work on the same number of pixels for each frame. However the thresholding algorithm can be modified to work on a sub-window of the total image, just in the same way the correlation algorithm is implemented. The reason why it is not done in the version of the thresholding algorithm presented here is because other performance criteria will justify the choice of the correlation algorithm in the research, and improvements on the thresholding method have not been implemented because they would not have been used directly for the project. However, the conclusion on the processing time analysis alone shows that the correlation algorithm performs better.

A quick comment is required here concerning the absolute time required by the correlation algorithm. There is a ratio of about 1 to 10 between video real time and the required processing time on a single target. This allows several targets and a few minutes of video data to be processed within half a day of the recording. It is important that the processing does not take too long, so the measurement system is practical. The time repeatability is one of the system design drivers pointed out in section 4.1. For most applications, a few minutes of motion measurements and 5 to 10 targets in the image are sufficient. Since the system can provide useful data within hours of the recording, the data can be analysed, adjustments can be made accordingly and a new recording session can be started at the latest one day after the original recording session. So the processing time is short enough to provide a measurement system with a high repeatability suitable for research involving iterative measurements.

Position accuracy

It is difficult to assess the accuracy of the position retrieval for real targets since their motion is not known *a priori*. One way to assess this accuracy in lab conditions is to track an object of known motion, for example an object in free fall or a pendulum. However, the accuracy of the tracking algorithms depends on the speed of the object in the field of view, as it affects its appearance. As stated before, a fast moving object can appear to be blurred in the image, and this blurring will affect the position at which maximum correlation is obtained, or the centre position of the remaining pixels after thresholding. For that reason, the accuracy of the algorithms, regardless of blurring effects or issues linked to the change of apparent shape of the target, is better assessed in slow motion conditions. Here, wheat moving slowly in the image is used for the accuracy assessment. Other issues concerning fast movements and shape changes are discussed in the next section.

When the wheat is moving slowly, the centre position of the target is easily followed in the image as it is marked with a green cross. Visual comparison of the motion of this cross as the wheat moves, after processing by the two algorithms, show that in both cases the position of the cross is stable with respect to the red target, but occasionally flickers by a maximum of two pixels.

With the thresholding method, the flickering is due to the fact that the pixels at the edges of the red target are at the limit of the colour range to match, and consequently they are not consistently categorised as being part of the target in all frames. The result is that the calculated centre position of the remaining red pixels can move slightly between consecutive frames. One way to reduce this effect is to increase the range of acceptable colour values in the thresholding process. By doing so, more pixels are categorised to match the colour of the label, which means that more “noisy” pixels appear in the image. The increased noise requires more low-pass filtering with the averaging filter, and consequently more processing time. There is a trade-off here between the desired accuracy of the position retrieval and the acceptable processing time. The final decision from this trade-off depends on the application.

With the correlation method, the correlation coefficients of the pixels in the neighbourhood of the centre of the label have very close values, since the mask matches well the target in the image when centred on these pixels. Consequently, the highest correlation can be obtained for a different pixel between consecutive frames.

The two pixel amplitude of the flickering is determined by visually analysing each frame of the video sequence with Adobe Premiere, and measuring the variation of the position of the green cross with respect to the true centre position of the label, as seen in the image. The standard deviation on this variation is of 2 pixels (obtained from a video sequence of 100 frames). It takes approximately the same value in both the row and column directions, and for the two tracking methods. A two pixel variation corresponds to an angular change of about $2.1.10^{-3}$ radians. This value is obtained from the video image calibration to be detailed in section 4.7. At a typical target-to-camera range of 1 metre, the corresponding position accuracy is 2.1 mm for a displacement in the image plane (i.e. perpendicular to the viewing axis).

Reliability and adaptability

The limits of the tracking software are reached in the case of fast target motion. A fast moving object in the scene appears blurred in the image because of the way the video image is recorded by the camera. In fact, for each frame, the camera records two half frames interleaved by line in the row direction. Consequently, when an object in moving fast in the field of view, the two half frames do not correspond to the object in exactly the same position, and the result is that adjacent lines in the frame do not appear to be properly aligned vertically. An example is shown in Figure 4.9(a) for a moving aircraft. In this case, both tracking methods will lose accuracy. For applications where the object is moving fast, the system presented here is not suitable. However, in the case of wheat motion measurements, the red labels placed in the wheat only appeared blurred in the most windy conditions of the entire data collection campaign. In this case, the position of the target in the frame is set to the coordinates (0,0) (the origin of the image) in the correlation algorithm, and in the next frame is processed. Post-analysis of the data allows to easily spot the frames where the position of the target could not be retrieved due to excessive blurring. Linear interpolation between the two neighbouring frames gives an estimate of the expected position of the target in the frame missed by the correlation algorithm.

The thresholding algorithm cannot fail to compute a target position even when it is blurred by its fast movement because there are always some pixels from the red label within the colour range to match. This can be viewed as a more robust method in case of high speed conditions, but in fact it is not because the centre position calculated by the algorithm depends on where the blurring occurs, and therefore is highly variable from one frame to the next. Experience shows that the accuracy of the thresholding algorithm is inferior to that of the correlation algorithm in the case of fast movements.

Targets are not accurately tracked when they are occulted by another object in the field of view. Typically this case occurs in a wheat field when a wheat leaf is in a direct line between the camera and the target. The same interpolation scheme described above can be applied here, for frames where the target position is not determined directly.

In general, the correlation algorithm is more adaptable than the thresholding algorithm for the retrieval of a target position. This is due to the fact that, with the help of the target selector (section 4.6.5), any object having a typical shape in the image can be tracked. The thresholding algorithm requires that the target in the image has a specific colour, almost unique in the image, and the colour values of its pixels need to be known as they define the colour to match. The correlation algorithm is more convenient to use in that respect since it only requires to select the area of interest in the first frame of the video film, and this area is tracked automatically for the entire video sequence. In fact, tracking with correlation algorithm was successfully performed on other shapes than the regular, uniformly red labels on the wheat elements (e.g. on the plane shown in Figure 4.9). The possibility to track objects of a non-uniform colour is specific to the correlation algorithm, which appears to be very adaptable in that respect.

Summary of the performance of the two algorithms

Based on the comments above, Table 4.1 gives performance coefficients to the two tracking methods, for the four following performance criteria: position accuracy, processing time, adaptability to different target types, performance on fast moving objects. The performance coefficients range from 1 (poor performance) to 5 (high performance).

	Position accuracy	Processing time	Shape adaptability	Fast moving object tracking
Thresholding	4	5	1	3
Correlation	4	2.5	5	4

Table 4.1: Performance coefficients of the two tracking algorithms (1:poor performance, 5: high performance)

The choice between one of the two algorithms depends on the application, and on how this application considers the relative importance of the four performance criteria. For well-defined targets having an easily recognisable colour in the image, and when a fast repeat time between measurements is required, the thresholding algorithm should be preferred. However, in the more general case where the object to track is not precisely defined, or can change slightly in shape and colour between experiments (due for example to a change of viewing angle of the camera or a change of illumination), then the correlation algorithm provides the necessary flexibility.

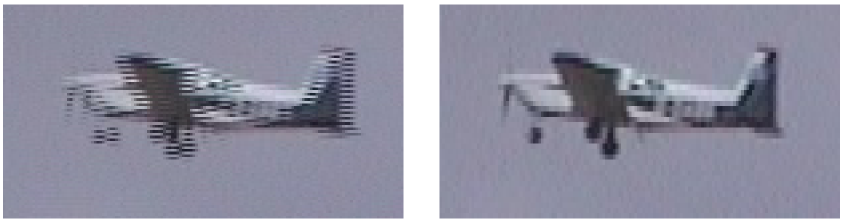
The correlation algorithm was used for the tracking of wheat because of this additional flexibility. In addition to this, providing the possibility to follow any given object in a video image is particularly interesting from a commercial point of view. Finally, the tracking by correlation can be significantly improved by additional processing techniques, as the next section will show.

4.6.8 Conclusion: potential improvements of the tracking algorithms

The description of the tracking would not be complete without giving a brief introduction to some potential improvements. These improvements have not been programmed in the algorithm because they are not the main focus of the research. The tracking system works efficiently for the measurement of wheat motion, which is the main concern here. However, additional processing techniques would improve its capability for a more general use. The possibility to access pixel values in a standard AVI file opens a wide range of processing techniques and manipulations on these values.

Un-interleaving of the video frames

It was explained in the previous section that lines around fast moving objects appear to be shifted from their adjacent lines due to the recording process of the video camera in half frames interleaved by line. For a given frame, each line with, say, an even number can easily be copied and pasted to the line directly below (with an odd number), and the line shift therefore disappears in the image, at the expense of a decreased resolution in the row direction. An example is shown in Figure 4.9 for a plane. The difference between before (Figure 4.9(a)) and after (Figure 4.9(b)) un-interleaving shows the values of the technique.



(a) Before un-interleaving

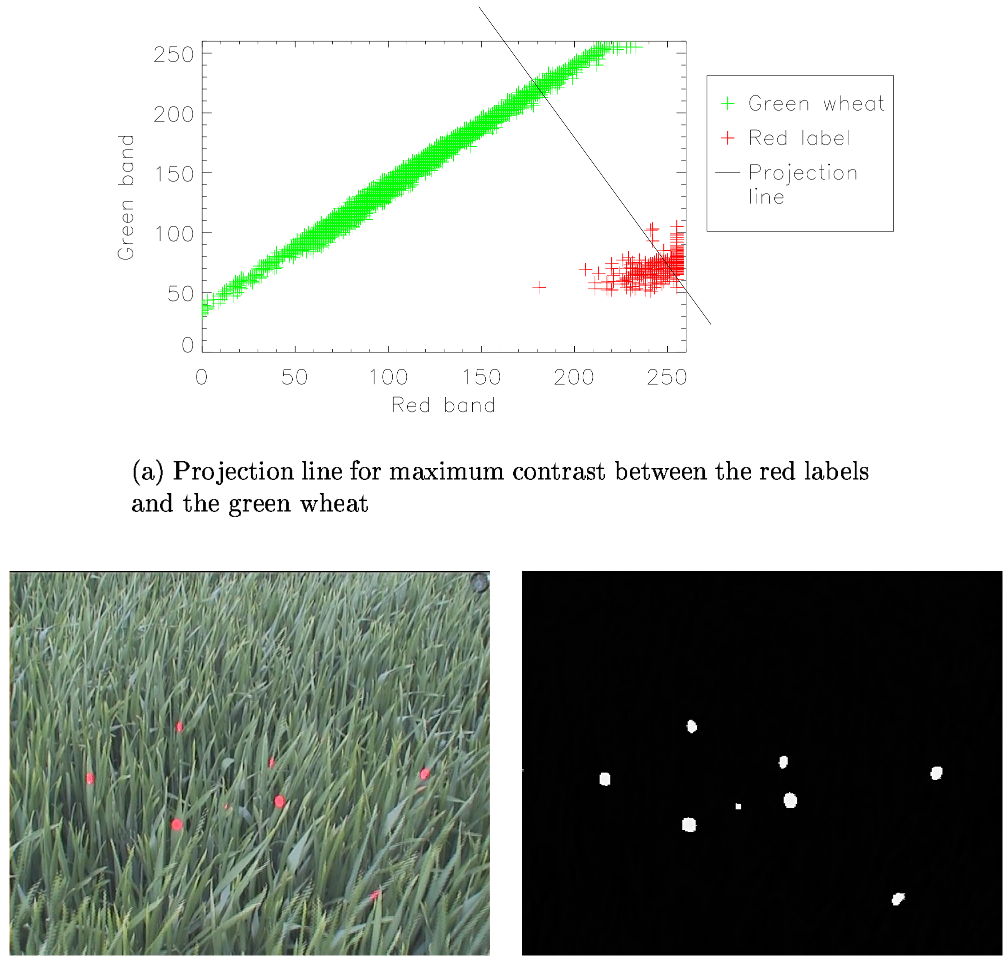
(b) After un-interleaving

Figure 4.9: Effect of un-interleaving of video frames on fast moving objects

For the tracking of fast moving object, preliminary un-interleaving would reduce the blurring effect, if not suppress it totally, and would therefore improve the accuracy of the position retrieval by the tracking algorithms.

Contrast enhancement

Both tracking techniques would benefit from an enhancement of the contrast between the object to track and the background. Returning to Figure 4.5, it is clear that the individual use of any of the three colour bands would not improve the contrast between the target and the wheat background, since the latter covers the whole range of colour values in all three bands. However, focusing for example on the green and red bands shown in Figure 4.5(a), it can be very valuable to take a linear combination of the green colour values and the red colour values for each pixel and to display the result on a grey scale. To justify this statement, Figure 4.10(a) shows the same pixels as Figure 4.5(a), but only for the red label pixels and the green wheat pixels. The coefficients of the optimum linear combination define a straight line in the green-red colour subspace, and they are chosen so that the regions occupied by the green wheat pixels and by the red label pixels are as far as possible when projected along the straight line. The techniques is applied to a video sequence of which Figure 4.10(b) shows the first image. The resulting image after linear band combination is displayed in Figure 4.10(c).



(a) Projection line for maximum contrast between the red labels and the green wheat

(b) Original image

(c) Result after the linear combination

Figure 4.10: Contrast enhancement by appropriate linear combination of the red and green band

The labels after the simple linear combination of two bands have a much higher contrast than in the original image. This example shows the value of band combination to facilitate the tracking. The simple linear combination of two bands may not always be possible for other applications, but it is in principle possible, by plotting graphs similar to that of Figure 4.10(a), to find a band combination improving the contrast. In the general case, the combination may require the use of the three colour bands and of non-linear combinations.

Adaptation to shape changes

In the correlation algorithm, the shape of the target to track is selected on a still image, usually the first frame of the video file, and the same mask is used for all the remaining frames. If the object to track changes of apparent shape with time, the correlation with its original appearance may not yield accurate results. A possible improvement of the correlation algorithm is to use a variable mask. For any given frame the correlation can be calculated with a mask created from the previous frame, centred on the precedent position of the target, and with the original mask size. With this method, small changes of the shape of the target from frame to frame are gradually taken into account.

The adaptable mask is not a requirement for the red labels placed on the wheat as their shapes is approximately constant during the entire video sequence. However, it can be useful to track objects in a more general case.

4.7 Video image calibration

In all the discussion so far, it was assumed on several occasions that the row and column number of a particular pixel in the image could be translated into azimuth and inclination angles to input in the model equations. It is the purpose of this section to present this calibration work. The output of this work is a couple of analytical functions giving the inclination and azimuth angles α and β, or more precisely the tangents of these angles, as a function of column and row number.

4.7.1 Methodology and experimental setup

The video camera lens distorts the image in a characteristic 'barrel' distortion. If no distortion was present in the image then the column and row number for a particular point would be proportional to the coordinates (x,y) of that point in a plane parallel to the image plane. With the introduction of distortion in the image, the relation is no longer simple and needs to be modelled. The modelling involves simple analytical functions relating the row and column number of a particular point to the inclination and azimuth with respect to the camera line of sight.

In order to calculate the coefficients of the analytical functions used for distortion modelling, the camera is placed on its tripod at a known distance from a reference grid which is imaged with the camera. The grid is composed of a series of perpendicular lines precisely ticked. It is positioned so that lines on the grid represent horizontal and vertical lines (see Figure 4.11). The vertical alignment is made with a plumb line. The camera itself is also carefully vertically aligned. The grid-to-camera distance is measured from the grid to the inside edge of focusing ring of the camera. Although in principle the ideal distance should be measured from the focal plane of the camera, in practice this is not required as long as the same reference point is always used during all work involving measurements with the camera.

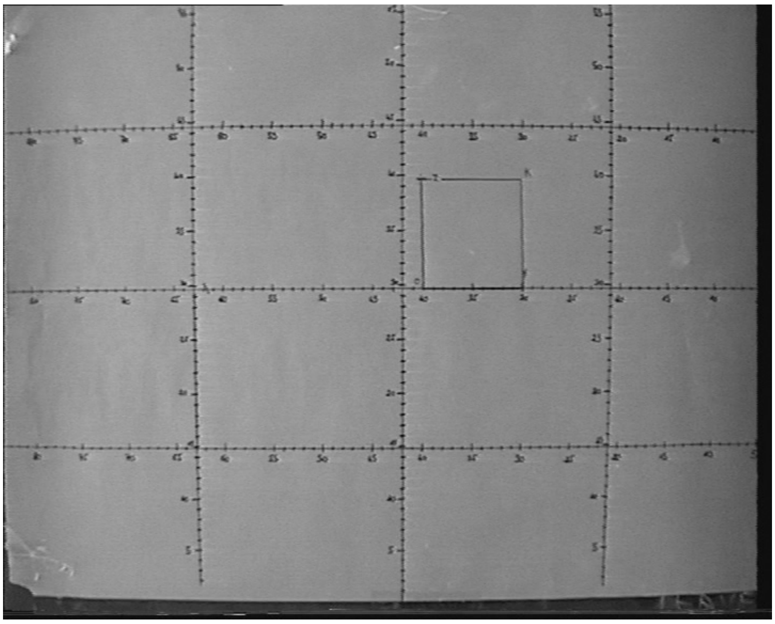


Figure 4.11: Grid used for the video image calibration

It is important to note that the grid images were taken with the zoom level set to its minimum. If a zoom is applied, the relation between row and column number and inclination and azimuth angles changes. Since the camera does not provide a quantitative value for the zoom level, it is recommended that it is always set to its minimum when measurements are made. The lines of the grid were carefully drawn so they form two sets of parallel lines, each set being perpendicular to the other. Figure 4.11 shows that parallel lines do not appear strictly parallel on the image due to the lens distortion.

The grid origin is at its bottom right corner and the image origin at the top left corner of the image. A common origin is set to be at the centre of the image, since the inclination and azimuth angles required are defined from this centre position. In the remainder of this document, the real coordinates of a point on the grid are noted (x_r,y_r) and correspond to the horizontal and vertical distance in mm from the image centre point. The image coordinates (x_i,y_i) are given in column number, row number respectively, from the same image centre.

The real coordinates of points on the calibration grid were noted simultaneously with their column, row number on the image. Then from these data points, a least square solution is found for the parameters of the analytic distortion functions tested.

4.7.2 Determination of the coefficients of the calibration function

Different functions were tested to fit the data points (x_r, y_r, x_i, y_i). The selected function is a 3^{rd} order polynomial. In order to assess the accuracy provided by the 3^{rd} order polynomial, the case where no distortion is accounted for will also be presented. Note that a wide range of other functions were tested to represent the distortion effect, but for conciseness only the selected 3^{rd} order polynomial is presented here. The quantities x_{est} and y_{est} are the coordinates of a target point in mm, estimated from the third order polynomial. Finding the appropriate coefficients of the polynomial is done by minimising the quantity:

$$\Delta = \sum_{j=1}^{N} \left[x_{r_j} - x_{est}(x_{i_j}, y_{i_j}) \right]^2 \tag{4.28}$$

Such a least square solution is found using the method detailed in [56], based on Singular Value Decomposition (SVD).

Third order polynomial model

The function used to model distances in mm in the horizontal direction is:

$$x_{est}(x_i, y_i) = ay_i^3 + bx_i^3 + cy_i^2x_i + dx_i^2y_i + ey_i^2 + fx_i^2 + gy_ix_i + hy_i + ix_i + j \tag{4.29}$$

And similarly, for the vertical direction:

$$y_{est}(x_i, y_i) = ky_i^3 + lx_i^3 + my_i^2x_i + nx_i^2y_i + oy_i^2 + px_i^2 + qy_ix_i + ry_i + sx_i + t \tag{4.30}$$

A least square data fitting method based on SVD [56] is available from the IDL language. The obtained coefficients a to t are summarised in Table 4.2 for the first camera, and in Table 4.3 for the second camera.

$x_{est}(x_i, y_i) = ay_i^3 + bx_i^3 + cy_i^2x_i + dx_i^2y_i$ $+ey_i^2 + fx_i^2 + gy_ix_i + hy_i + ix_i + j$	$y_{est}(x_i, y_i) = ky_i^3 + lx_i^3 + my_i^2x_i + nx_i^2y_i$ $oy_i^2 + px_i^2 + qy_ix_i + ry_i + sx_i + t$
a=1.82.10^{-8}	k=1.33.10^{-7}
b=2.05.10^{-7}	l=7.69.10^{-9}
c=1.97.10^{-7}	m=-5.32.10^{-9}
d=1.02.10^{-9}	n=1.85.10^{-7}
e=9.73.10^{-7}	o=2.96.10^{-5}
f=-2.87.10^{-5}	p=5.93.10^{-6}
g=2.85.10^{-5}	q=-2.41.10^{-5}
h=-1.17.10^{-2}	r=0.79
i=0.87	s=-3.79.10^{-3}
j=0.84	t=-0.32

Table 4.2: Coefficients of the 3^{rd} order polynomials, obtained for the first camera, relating the positions x_{est} and y_{est} on the calibration grid to the column and row number in the image (x_i and y_i respectively

Note that the fact that the high order term coefficients are small does not necessarily mean they are insignificant. In fact only comparison of the terms of the same order is directly possible.

$x_{est}(x_i, y_i) = ay_i^3 + bx_i^3 + cy_i^2 x_i + dx_i^2 y_i$ $+ey_i^2 + fx_i^2 + gy_i x_i + hy_i + ix_i + j$	$y_{est}(x_i, y_i) = ky_i^3 + lx_i^3 + my_i^2 x_i + nx_i^2 y_i$ $oy_i^2 + px_i^2 + qy_i x_i + ry_i + sx_i + t$
a=-3.71.10^{-8}	k=3.27.10^{-7}
b=3.38.10^{-7}	l=7.92.10^{-9}
c=3.40.10^{-7}	m=-1.20.10^{-8}
d=2.50.10^{-8}	n=2.91.10^{-7}
e=-3.90.10^{-6}	o=-5.27.10^{-5}
f=8.91.10^{-6}	p=1.30.10^{-5}
g=-7.07.10^{-5}	q=7.79.10^{-6}
h=-2.22.10^{-3}	r=1.14
i=1.25	s=-3.03.10^{-4}
j=1.04	t=-1.15

Table 4.3: Coefficients of the 3^{rd} order polynomials, obtained for the second camera, relating the positions x_{est} and y_{est} on the calibration grid to the column and row number in the image (x_i and y_i respectively

Model with no distortion

If no distortion is accounted for in the image, the column and row number x_i and y_i are proportional to the grid coordinates x_r and y_r respectively. This is an important case to model as it allows comparison with the previous case. It can be used to see if the 3^{rd} order polynomial truly provide an improvement.

In the case of no distortion:

$$x_{est} = ax_i \qquad y_{est} = by_i \tag{4.31}$$

The coefficients a and b are simply found by averaging $\frac{x_r}{x_i}$ and $\frac{y_r}{y_i}$ respectively. The values found for a and b for camera 1:

$$a = 0.90 \qquad b = 0.80 \tag{4.32}$$

Similarly, for camera 2:

$$a = 1.21 \qquad b = 1.15 \tag{4.33}$$

These coefficients are comparable in magnitude to the coefficients i and r of Tables 4.2 and 4.3. This shows that the relation between real coordinates and image coordinates is mainly linear as expected: the distortion is a small effect added to the perfectly linear case.

4.7.3 Angle retrieval

The inclination and azimuth angles α and β for a point on the image are calculated from the values of x_{est} and y_{est} derived from the functions specified in the previous sections:

$$\tan \alpha(x_i, y_i) = \frac{y_{est}(x_i, y_i)}{\mathcal{D}} \qquad \tan \beta(x_i, y_i) = \frac{x_{est}(x_i, y_i)}{\mathcal{D}} \tag{4.34}$$

where $\mathcal{D}$ is the distance measured between the grid and the camera, as defined in section 4.7.1. $\mathcal{D}$=832 mm for the first camera and $\mathcal{D}$=1204 mm for the second camera. Since the algorithms used for target position retrieval require only the *tangent* of the inclination and azimuth angles, there is no need to invert Equation (4.34) to obtain the actual value of α and β.

4.7.4 Accuracy analysis

The average error ε_α and ε_β in the estimation of the angles is calculated as follows:

$$\begin{aligned} \varepsilon_\alpha &= \sqrt{\overline{\left(\widehat{\tan}\,\alpha - \tan\alpha_{true}\right)^2}} \\ \varepsilon_\beta &= \sqrt{\overline{\left(\widehat{\tan}\,\beta - \tan\beta_{true}\right)^2}} \end{aligned} \tag{4.35}$$

Here $\widehat{\tan}\,\alpha$ is the estimated value from Equation (4.34), $\tan\alpha_{true}$ is the true angle derived from the measured positions on the grid, and the bar denotes averaging. So the error is defined as the square root of the average square difference between the measured and modelled tangent of the angle. The average is calculated over all the data points used to calculate the fitting function. The same definition applies to ε_β.

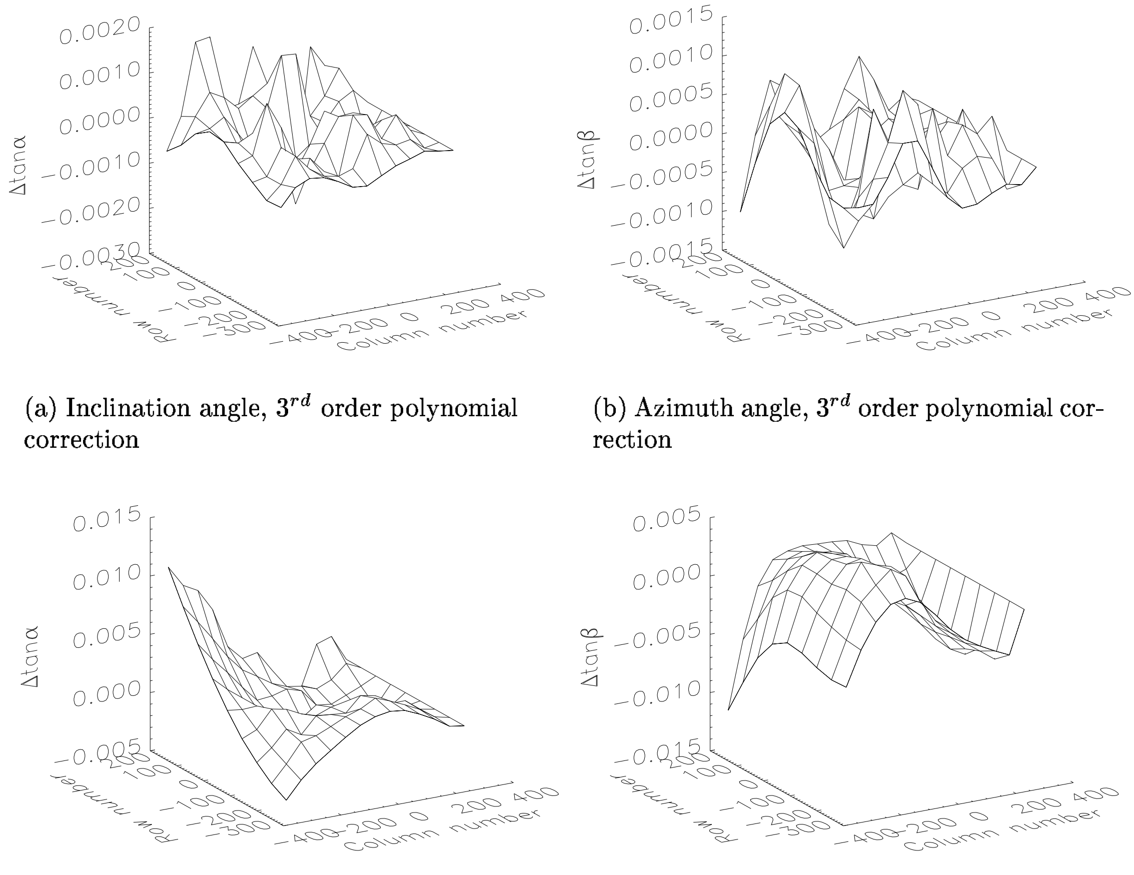
For quantitative analysis the values of ε_α and ε_β are compared to the average change in $\tan\alpha$ and $\tan\beta$ inferred by a displacement of one pixel in the column and row directions respectively. For camera 1 ($\mathcal{D}$=832 mm), one pixel in the row direction corresponds on average to $\tan\alpha = 9.67.10^{-4}$ and one pixel in the column direction corresponds to $\tan\beta = 1.07.10^{-3}$. For camera 2 ($\mathcal{D}$=1204 mm) these values are $\tan\alpha = 9.66.10^{-4}$ and $\tan\beta = 1.07.10^{-3}$. Values are very similar for the two camera because they are of the same model.

	ε_α (absolute value)	ε_α (in pixels)	ε_β (absolute value)	ε_β (in pixels)
Camera 1				
3^{rd} ord. pol.	$6.06.10^{-4}$	≈ 0.63	$8.94.10^{-4}$	≈ 0.83
No distortion	$2.46.10^{-3}$	≈ 2.54	$3.17.10^{-3}$	≈ 2.95
Camera 2				
3^{rd} ord. pol.	$4.26.10^{-4}$	≈ 0.44	$4.45.10^{-4}$	≈ 0.41
No distortion	$1.41.10^{-3}$	≈ 1.46	$2.18.10^{-3}$	≈ 2.03

Table 4.4: Root mean square error on the 3^{rd} order polynomials giving the tangent of the inclination and azimuth angles as a function of the row and column number in the image

Table 4.4 shows that the 3^{rd} order polynomial distortion function provides an improvement compared to the no distortion case. It is expected that a 4^{th} order polynomial would improve the accuracy. However it has not been used because higher order polynomials would start fitting the measurement noise on the calibration data points. The accuracy of the 3^{rd} order polynomials is less than a pixel, which matches the accuracy of the measurement of the data points used for the calibration.

The last study made on the accuracy concerns the repartition of the error throughout the image. For that purpose the difference ($\Delta\tan\alpha$ and $\Delta\tan\beta$) between the true values of $\tan\alpha$ and $\tan\beta$ and the estimated values $\widehat{\tan}\,\alpha$ and $\widehat{\tan}\,\beta$ is calculated at each point of measurement on the calibration grid. The results are plotted in Figure 4.12 for the first camera.



(a) Inclination angle, 3^{rd} order polynomial correction

(b) Azimuth angle, 3^{rd} order polynomial correction

(c) Inclination angle, no distortion correction

(d) Azimuth angle, no distortion correction

Figure 4.12: Difference between the actual and estimated tangent of the inclination and azimuth angles, plotted for all points of the calibration grid

It is visible from Figure 4.12 that the 3^{rd} order polynomial corrects from the effect of image distortion, particularly at the edges of the field of view. The remaining errors after correction are of the same order of magnitude or smaller than the measurement noise. A similar result is obtained for camera 2.

4.7.5 Conclusion

The distortion functions, from which the tangents of the inclination and azimuth angles are derived, were calculated in this section. Their accuracy is about 1 pixel.

The main advantage of the video system used at Cranfield is its low cost, and the price to pay for this low cost (using commercially available digital cameras) is on the quality of the optics. There is a trade-off to be made between the accuracy required for the target position/motion measurement and the competitiveness of the system with respect to cost and manoeuvrability. For crop motion measurement, the balance goes in favour of the latter.

4.8 Practical implementation

The entire measurement system has now been described in detail in the previous sections and it now time to explain how it is used in practice for field measurements. The different points made in this section are the result of a number of repeated experiments and reflect the experience gained in these experiments. It is the aim of this chapter to share this experience with readers who may want to reproduce the measurement setup. The comments are made for the specific case of wheat motion measurements.

4.8.1 Positioning of the camera and of the red labels

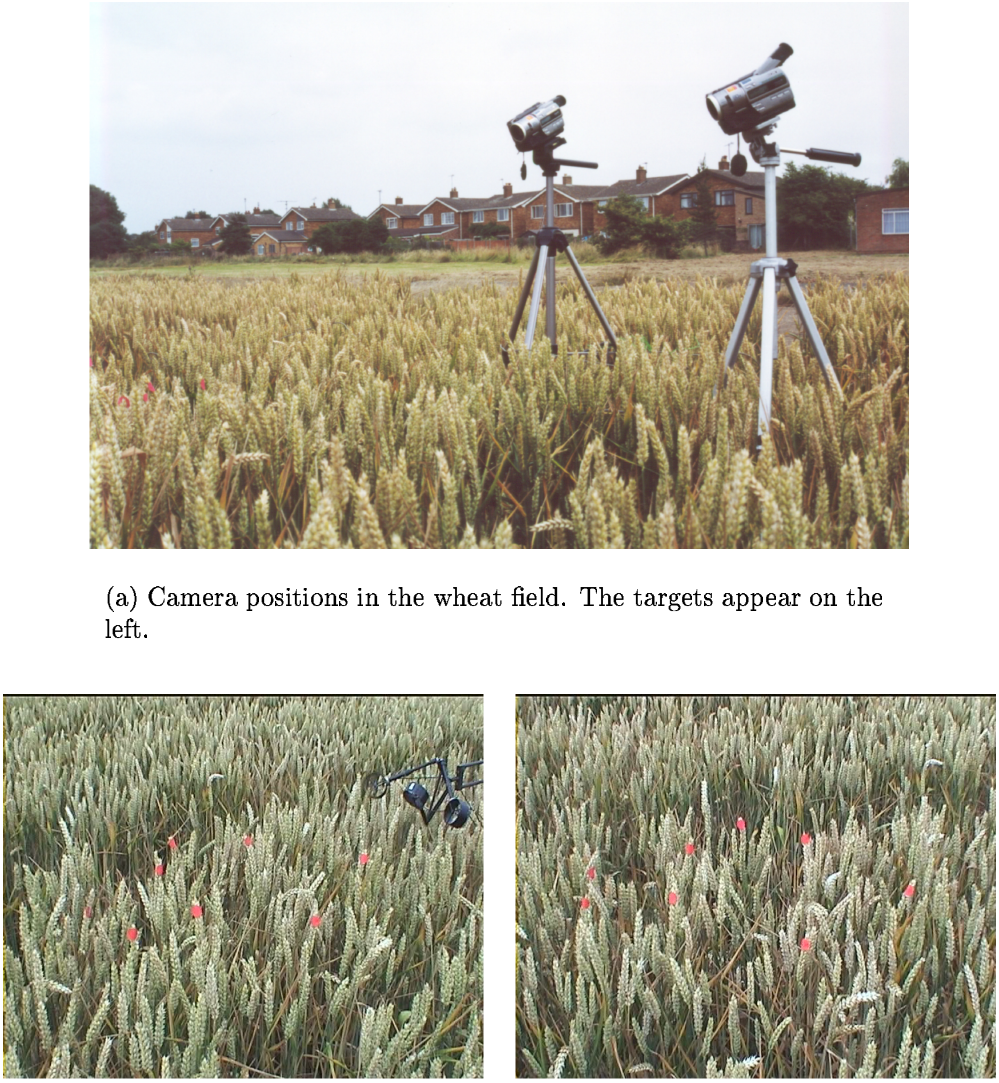
The optimum position of the cameras with respect to the target depends on several factors. First the distance to the targets must be sufficient short to make the targets easily visible in the image, but sufficiently long to incorporate several targets in the field of view. In the case of wheat, the labels glued to the wheat have a diameter of 20 mm. Such a size is compatible with the dimensions of the wheat elements. At a distance of about 1 to 1.5 m, the labels occupy approximately 30 pixels along their diameter, which is convenient for the tracking.

The field of view of the cameras can be calculated using the calibration functions derived in section 4.7. The video image containing 720x576 pixels, the camera field of view is about 43°x31° (azimuth x inclination). Assuming that the camera-to-target distance is 1 m at the centre of the image, and that the cameras are pointing down at an angle of about 30° with the local horizontal, then the video image covers a ground area of 0.8 m in the camera azimuth direction and of 1.4 m in the inclination direction. These values are rough estimates only, but they are useful to relate to the measurement requirement of section 3.6, stating that several targets should be measured simultaneously for spatial correlation analysis. They show the spatial extent over which the targets can be tracked. The value of 1.4 m in the inclination direction is in practice reduced because targets at the far range of the image would appear too small to be tracked accurately.

There is a trade-off to make between the spatial coverage of the cameras and the accuracy of the tracking. With a long camera range, the spatial coverage of the cameras is increased but the position measurement is less accurate. The opposite occurs at short ranges. A distance of about 1.5 m to the middle of the measurement volume seems to be a good compromise in the case of wheat motion measurements.

In theory the distance between the two cameras can be chosen arbitrarily. In practice however, since the red labels have their face pointing towards one direction, the cameras should be placed so that the labels to track are easily visible by both. A distance of about 1 to 2 m between the cameras is appropriate.

Figure 4.13(a) shows the positioning of the cameras with respect to each other and to the targets to track. Figures 4.13(b) and 4.13(c) show the view that the two cameras have of the targets in the configuration shown in Figure 4.13(a).



(a) Camera positions in the wheat field. The targets appear on the left.

(b) View from the left camera

(c) View from the right camera

Figure 4.13: Camera positioning for wheat motion measurements.

It can be seen from the images of the wheat shown in Figure 4.13(b) and 4.13(c) that the motion of the wheat leaves is virtually impossible to measure with the measurement system here since they do not appear clearly on the image, being too deep inside the vegetation. The wheat heads however, can be easily isolated and measured. Early in the season, when the wheat leaves only are present in the vegetation, their motion could be measured. Later in the season, as in the case of the pictures of Figure 4.13, only the wheat head motion was measured. This limitation will have implications for the modelling of the coherence. They will be discussed in Chapter 7.

4.8.2 System calibration

Having positioned the cameras and the red labels, the next step is to perform the system calibration, as detailed in section 4.3.4. The reference frame placed in the field of view is shown in Figure 4.14. Four points are used for the calibration. The origin of the reference system is set to be at the origin of the reference frame. The calibration points are chosen on each of the three axes of the frame. These axes define the fixed reference system for the measurements. They have been precisely ticked so the position of the calibration points is accurate. The three axes of the wooden frame are not exactly perpendicular to each other, but the angles between them was measured and taken into account in the calibration.

The reference frame is not needed at all times in the field of view, provided the cameras are not moved during the recording. In the case of wheat measurements it is even required that the reference frame is removed after it is recorded for a few seconds by the two cameras, since it would hide some of the targets and impinge the free motion of the wheat.

As stated in section 4.3, the calibration points should cover most of the field of view if possible, in order to get maximum accuracy in the calibration. The reference frame of Figure 4.14 is suitable in that respect since different points along the three axes can be chosen, depending if they appear in the field of view. The calibration procedure described in section 4.3.4 is performed indoors, after download of the video data to the computer.

4.8.3 Time synchronisation

Synchronisation between the two video cameras is necessary in order to know the position of a given target in the two images at the same time. Since the two cameras cannot be started exactly simultaneously, it is necessary to provide to the cameras an external signal short enough so it can be detected in a single frame. The frame in which this signal appears in the two images indicates a common point in time in the two video sequences. The two video cameras are synchronised by noting the frame number of the appearing signal in the two video films. In practice the signal is given by a classic camera flash, pointed at the two video cameras. The camera flash is short enough in time to appear in a single frame, and is therefore perfectly adapted to provide a good synchronisation signal. Several camera flashes can be used at different instants in order to check that the video cameras remain synchronised throughout the recording. Experience here shows that they do record video data at exactly the same rate, over periods of time of at least 30 minutes.

4.8.4 Recording phase

After synchronisation the cameras are ready to record useful data. The recording length is in principle only limited by the available length left on the Digital 8 tapes (maximum of 60 minutes on a blank tape) or by the charge left on the battery. In practice, it is found useful to use the free time available during the recording to measure experimental parameters. In particular, the distance between the video cameras is a valuable parameter as it can be used to check the accuracy of the system

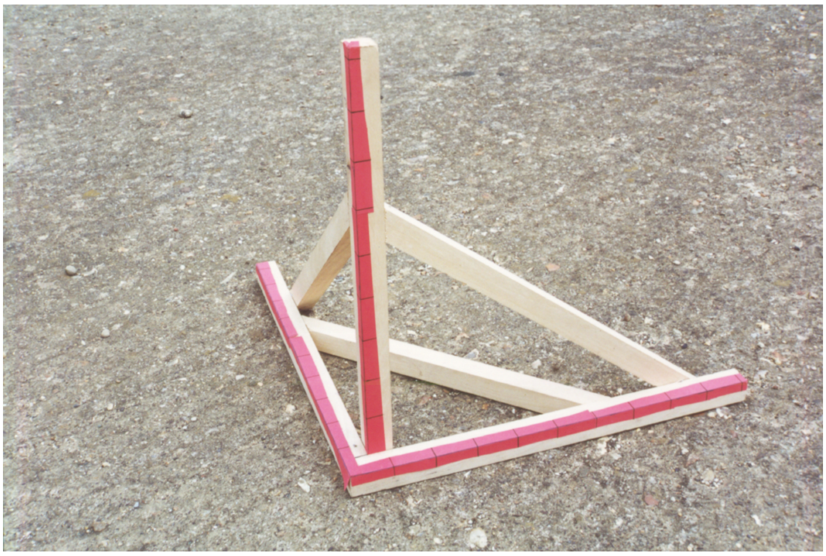


Figure 4.14: Reference frame used for system calibration

calibration. Other useful parameters include the crop height, and parameters related to the wind data recording (to be discussed in Chapter 5). For the wheat motion measurements, a typical recording sequence lasted between 20 and 30 minutes.

4.8.5 Post-recording processing

The remaining work after recording is done indoors. The video data from the two camera is downloaded to the PC. Due to the 2 Gbyte limit on the video files, only about 10 minutes of QuickTime film can be stored in a single file, so it is important to download the part of the data which is most interesting for the purposes of the research. In the case of wheat motion, moments of high winds are preferred.

It is important to note here that there is also a 2 Gbyte limit on the AVI file. The AVI format required for processing being uncompressed, the maximum time length of a single AVI file at 25 frames/s is about 67 seconds. In the measurements presented here, the AVI files are set to cover 1 minute of video data.

The Quicktime films obtained from Moto DV are imported into Adobe Premiere. Then the frame in which the camera flash appears is noted. It is used to export the same 1 minute sequence of interest from both video films into the uncompressed AVI format.

A single frame containing the reference frame is extracted from the two films, and used for calibration. Masks around the targets to track are created with the target selector, and the tracking program can then be started. With the output text files containing the row and column number of the targets, and with the camera parameters from the calibration, the coordinates of each target in the reference system, defined by the frame of Figure 4.14, is retrieved. These coordinates are those of the vector $\mathbf{p}'$ introduced in section 4.2. The point P_0 is chosen to be in the region surrounding the targets, since it is required that it should be close to them, inside the measurement volume. Only the positions relative to P_0 are of interest in the case of wheat motion anyway, since it is the amplitude and direction of the motion which is interesting for comparison with the wind data.

4.8.6 Total time of the experiment

It is useful to give here a time breakdown of the entire experiment in order to judge on its repeatability, pointed out as one of the design drivers. The time needed for the different parts of the measurement process are (the values are approximate):

- Installing the cameras and the red labels on the wheat heads: 10 minutes.
- Recording the reference frame for calibration: 2 minutes.
- Time synchronisation with the camera flash: 2 minutes.
- Recording of useful wheat motion data: 25 minutes.
- Un-installing the cameras: 5 minutes.
- Video data download with MotoDV: 2x15 minutes = 30 minutes.
- Finding the common camera flash in the two films: 2x5 minutes = 10 minutes.
- Selection of a sequence of interest in the video sequence: First film: 10 minutes, second film (same area): 5 minutes. Total: 15 minutes.
- Export to the uncompressed AVI format: 2x15 minutes = 30 minutes.
- Tracking on a 1 minute video sequence: 10 minutes per target per film. Assuming 5 targets, the total is 100 minutes.
- System calibration: 15 minutes.
- Calculation of the target coordinates from the tracking output files: 5 minutes per target. 25 minutes for 5 targets.

The total time of the measurement process is about 45 minutes in the field, and 225 minutes (3h45mn) for post processing. The system is repeatable on a daily basis if required, which meets the original repeatability requirement.

4.8.7 Building the wheat motion database

Going back to the objectives of the research enumerated in section 2.5.2, one of the purposes of the wheat motion measurements is to provide a useful database that can be used for different issues related to SAR backscatter modelling and wind/crop interactions. The measurements were narrowed down to the case of wheat as the important point here is to demonstrate the feasibility of the measurement method.

It was shown in Chapter 3 that motion information is required on the different elements of the wheat, namely the wheat heads, the stalks and the leaves. However, because the cameras need to be in view of a clear target fixed to the element to track, it is not possible to measure reliably the motion of wheat elements which are too deep inside the vegetation. For that reason, the measurements were focused on the top layer of the canopy. At an early growth stage, the motion of the top leaves was measured with the system. At a later stage, the red labels were placed on the wheat heads, as they are the only elements clearly visible at all times in the canopy. The implications of having selected a certain type of wheat element are discussed in the next section (4.9).

Wheat motion measurements were taken in the growth season 2000 on 8 occasions from May 1^{st} to August 15^{th}. Details on the exact dates of the measurements are given in Chapter 6. On each measurement session the position of several targets in the measurement volume is measured, in order to increase the statistical significance of the motion data and for spatial correlation analysis.

The data acquisition, processing and analysis were all performed in spring/summer 2000, i.e. at the very end of the completion date of the research. It is the personal view of the author that the research would greatly benefit from an additional measurement campaign in 2001, covering different crop types, and with a higher measurement frequency, in particular in spring, at maximum growth rate. It was shown in the previous sections that the measurement system is capable of handling different crop configurations and can achieve a high repeatability. However, it could not be exploited at its full potential because of the constraints on the project's timescale.

Building the wheat motion database in 2000 has shown that the system does answer the necessary measurement requirements. It is hoped that it will be further exploited for more detailed analysis of the motion of crops.

4.9 Conclusion

4.9.1 System summary

The three-dimensional position of moving targets is measured here with two digital video cameras pointing towards the same areas at two different angles. The system is based on consumer equipment, as the two cameras are widely available in most video retailer shops. The computer and the interface card are also commonly available. It is possible to use such a system based on consumer electronics because the video data can be processed in its AVI format. Targets of any given shape contrasting with the background can be tracked and their position measured. The calculation of the object's three-dimensional coordinates is possible from the two viewing angles provided by the cameras.

4.9.2 System accuracy

In section 4.3 the accuracy of the system calibration was given by Equations (4.15) and (4.3), in terms of the radial and longitudinal accuracies of the camera position with respect to the calibration targets. Now that the capabilities of the system have been described, it is possible to provide a numerical estimate of these accuracies. For a target-to-camera distance of 2 m, and a typical distance between the calibration points of 30 cm, the accuracy on the camera position of the order of 1 cm (based on a camera angular resolution $\delta\theta \approx 0.0005$ radian, as derived from the image calibration, section 4.7). The Levenburg-Marquadt algorithm used in the calibration (see section 4.3) also provides an accuracy estimate based on the camera angular resolution, and also yields an accuracy of about 1 cm. Consequently, the absolute positions of the targets to track are also retrieved with this accuracy of 1 cm. However, the positions of a target *relative* to, say, an initial position are retrieved with an accuracy only limited by the accuracy of the tracking algorithm. It was shown in section 4.6.6 that this accuracy is of about 2 mm for displacements in the image plane, with a measurement system in the configuration suitable to wheat motion measurements. It is this 2 mm accuracy which sets the limits on the potential use of the motion data.

For the coherence model, the accuracy required was quoted to be 1.3 mm in slant range. It was shown in section 3.6 that, for a wheat motion mainly in a horizontal plane and a radar incidence angle of 23°, the required measurement accuracy is 3.3 mm. The system does provide the necessary accuracy in this case, but not necessarily in all motion situations and radar configurations. In fact, the accuracy of the retrieval also depends on the relative position of the two cameras, so it is different for all recording sessions performed in summer 2000. The accuracy of the measurement system will be illustrated in section 6.5.1 for a particular date.

Improvements on the motion retrieval accuracy can be made by reducing the camera-to-target distance, at the expense of a reduced field of view and increased blurring effects on the image due to fast target motions. It is decided to keep the system in its current configuration for the wheat measurements, as it provides a good balance between position accuracy, number of targets in the field of view, and size of these targets on the image for reliable tracking.

The motion of wheat is actually restricted to the motion of some of its elements, situated at the top of the canopy. For a fully grown plant the motion of the wheat heads was measured. However, the modelling presented in Chapter 3 showed that motion information on all elements of the plants are useful. So it is important to assess the limits imposed here by the measurement system. The wheat stalks are directly related to the wheat heads, so their motion can be retrieved from that of the wheat heads. This will be done by using an estimate of the wheat deflection, derived in Appendix G and discussed in more detail in Chapter 6. The leaves pose more problems since their motion is both related to that of the wheat stalks and also subject to direct wind forcing. For that reason, the statistical description of the motion of leaves will be based on analytical distributions rather than directly on the motion measurement data. However, the leaves motion at an early growth stage was measured directly and can be used to model coherence from early wheat.

The measurement system shows here one of its limits, since it cannot accurately retrieve the complex motion of leaves at a later stage of the growth. However, the amplitude of the leaves motion can be based on that of the wheat stalk, since they are physically related to it. Further discussion on this point will be presented in Chapter 7.

4.9.3 Critical discussion of the system's capabilities

The early design drivers set for the system were pointing out to cost, portability, and repeatability as three important goals to achieve. The overall cost of the system includes the two video cameras, the interface card, the computer and the editing software. With current prices for the cameras and the computer, the total cost amounts approximately to £2000. Most 3D measurement systems are especially built for a specific application and do not use consumer electronics. Consequently their price is usually much higher than that of the system described here. So the requirement on low cost is certainly met.

On the portability, again the system provides a very high performance since it only requires portable cameras and some calibration equipment. For that reason it is easily setup and can be configured in the most adapted geometry suitable for the application of interest.

The data processing can take less than a day, if short time sequences of a few minutes at most are processed. This is the case for wheat measurements, but for applications requiring longer video sequences, the processing time can be longer. It is the repeatability requirements of the application, coupled with the required frame rate and the required number of targets which determine if the measurement system is suitable from the point of view of its time repeatability.

Going back the objective 2 of the research, stated in section 2.5.2, the system is suitable for the measurement of wheat motion, from the cost, portability, and repeatability points of view. The limitations of the system concern its limited capability to measure motions inside the canopy. However they do not impinge on the main objectives of the research, which are to show that a useful crop motion database can be created by the measurement system and applied to the understanding of SAR coherence.

The possibility to access the video data in its AVI format opens to a wide range of processing techniques useful to improve the quality of the motion retrieval. In particular it was shown how the frame un-interleaving is potentially a useful pre-processing technique to track fast-moving objects. And a colour analysis of the target compared to the surrounding background can be used to determine the most suitable colour band combinations that can improve the target contrast. Other processing techniques are currently being investigated at Cranfield University, aiming to push further the capabilities of the tracking software and of the 3D motion retrieval. It is felt that the basic system built for the purposes of this research has a good potential for further developments and other applications.

4.9.4 Potential applications

Other applications of the 3D measurement system are currently being investigated at Cranfield University. Among them, the following seem to have a good potential:

- Measurements of the trajectory of planes during landing and take-off for training pilots.

- Space applications: measurement of the flatness of solar panels after deployment, determination of the relative attitude of neighbouring satellites in a constellation.

- Medical research: Measurement of the motion of injured persons after a recent operation involving motor members (e.g. broken leg, arm, etc...) or of elderly people with bone deficiencies.

In fact, Cranfield University has been contacted during the course of this research to investigate the possibility to develop the measurement system presented here as a commercial product. The side of the research presented in this chapter constitutes in itself an original part which can contribute significantly to different areas of research. It is possible to develop a measurement system as described here due to the recent availability of the technology, involving consumer products. It would not have been possible to develop the same product a few years ago, without the availability of digital cameras and the limited performance of personal computers.