Model Based Tracking for Navigation and Segmentation

B Southall[1,2], J A Marchant[1], T Hague[1], and B F Buxton[2]

[1] Silsoe Research Institute, Wrest Park, Silsoe,
Bedfordshire, MK45 4HS, UK

[2] Department of Computer Science, University College London, Gower Street,
London, WC1E 6BT, UK
B.Southall@cs.ucl.ac.uk

Abstract. An autonomous vehicle has been developed for precision application of treatment on outdoor crops. This document details a new vision algorithm to aid navigation and crop/weed discrimination being developed for this machine. The algorithm tracks a model of the crop planting pattern through an image sequence using an extended Kalman filter. A parallel update scheme is used to provide not only navigation information for the vehicle controller but also estimates of plant position for the treatment system. The algorithm supersedes a previous Hough transform tracking technique currently used on the vehicle which provides navigation information alone, from the rows of plants. The crop planting model is introduced and the tracking system developed, along with a method for automatically starting the algorithm. In applications such as this, where the vehicle traverses unsurfaced outdoor terrain, "ground truth" data for the path taken by the vehicle is unavailable; lacking this veridical information, the algorithm's performance is evaluated with respect to human assessment and the previous row-only tracking algorithm, and found to offer improvements over the previous technique.

1 Introduction

An autonomous agricultural vehicle [2] has been developed at the Silsoe Research Institute to perform the task of *plant scale husbandry*, which aims, for example, to reduce the use of chemicals in crop protection by treating individual plants and weeds separately, with little waste chemical sprayed onto the bare earth. Such a level of treatment provides obvious environmental and economic advantages over more traditional field spraying techniques. It is important in such applications both to be able to steer the vehicle accurately along rows of plants and to be able to identify where individual plants are.

The algorithm described here uses perspective images captured from a camera mounted on the front of the vehicle to provide estimates of the position of both the structure of the crop row planting pattern and the position of individual plants within that structure. At the heart of the algorithm is a Kalman filter with a non-linear measurement model (to correct for the perspective distortion of the images), which is used to track a model of the crop planting pattern through the image sequence generated as the vehicle traverses the field. The crop row structure is a cue used by the vehicle, in combination with non-vision sensors [3], to navigate along the rows. A Hough transform algorithm for tracking the plant rows has been previously reported [7] [8] and is currently part of the system. The existing algorithm extracts only the direction and offset of the rows of plants whereas the new algorithm also provides an estimate of individual plant positions which can then be used to target treatment by the spray system. A second advantage of the new algorithm is that the Kalman filter provides a covariance matrix for the state estimate which can be given to the vehicle controller; the Hough transform does not produce these covariances.

The use of a rigid model in tracking systems is well established, notably [4] where the rigid model in question consisted of control points on straight edge segments, thus suitable mainly for man-made *objects.* The problem here is extracting man-made *structure* imposed on a natural world. The rigid model gives a perfect version of the relationship expected between the plant centres on the ground plane, and the non-ideal nature of the real world must be catered for. However, how to model the effects of the noise of individual plant locations on the estimation of the crop planting pattern position is unclear; for the purposes of this paper the simplifying step of accommodating plant position noise as uncertainty in the observation model has been taken.

Other vision work within the scope of this project [10][11] (neither of which have been implemented on the vehicle) has addressed tracking individual plants rather than the planting pattern as a whole as approached in this paper; by tracking the whole pattern it is believed a more robust system will be realised. The two methods mentioned make use of plant models (in [11] a cluster of chain-coded areas, in [10] a more complex shape model) both of which could potentially be distracted by large clusters of weed material. By using the information available on the planting pattern, the search for plant material is constrained, and weed patches away from the crop structure will be ignored.

This paper describes the model of the planting structure, the extended Kalman filter used to track the model through the image sequence, and a method for automatically boot-strapping the algorithm. Finally, results are presented from experiments with real images (in the examples given, the crop is cauliflower) and conclusions drawn on the performance of the technique.

2 The Crop Planting Model

2.1 Model Parameters

Figure 1 shows a schematic view of a patch of crop, with the plants being represented by black circles. There are two sets of axes in the figure, (x_w, y_w) and (x_c, y_c, z_c) which represent the world and camera co-ordinate systems respectively, with the world z_w axis projecting out of the page, and camera axis z_c projecting into the page. It can be seen that the world y axis is coincident with the middle plant row. The model of the crop consists of the four measurements; r, the spacing between the rows, l the space between plants in the row, D_{-1}, which is a measure of the offset between the left-hand row of plants and the central row of plants and finally D_1, which is the corresponding quantity for the right-hand row. The model makes the assumption that the crop is planted in perfect position, although in reality there is error in the planting positions, and some plants are missing – both of these problems are tackled by the Kalman filter.

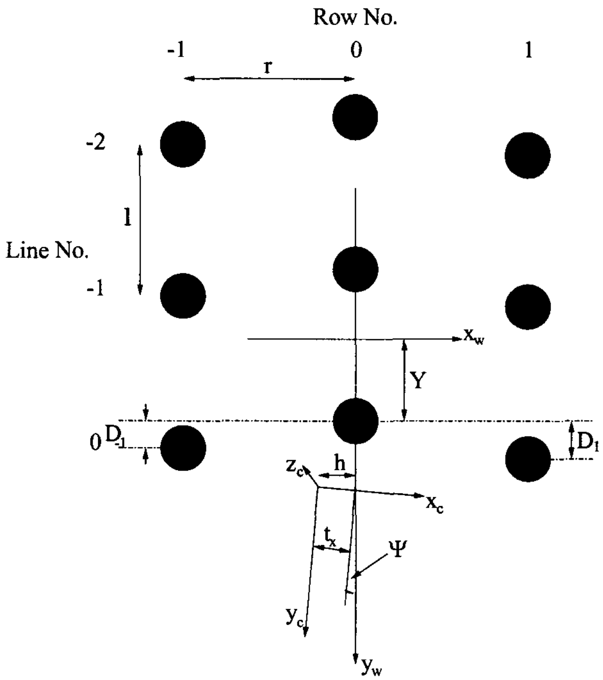

Fig. 1. The crop planting model

Three parameters (t_x, Y and Ψ), specify the position of the model relative to the camera as shown in the diagram, and these will later be seen as the state vector $\mathbf{x} = [t_x, Y, \Psi]^T$ estimated by the Kalman filter. The measurement Y is the offset in world co-ordinates from the world origin of the plant in the central row at the bottom of the image. The offset of the camera y_c axis from the world origin t_x is approximately equal to the distance h when angle Ψ is small. It can be seen then that the t_x and Ψ parameters may be used to provide navigation information in terms of a heading angle and offset of the camera (and hence the vehicle) relative to the rows, and the parameter Y, in conjunction with model parameters D_{-1} and D_1 can yield the position of individual plants via equations 1 and 2 (extended from those presented in [8]) below.

$$x_w = nr \tag{1}$$

$$y_w = ml + Y + D_n \tag{2}$$

The quantities $n \in \{-1, 0, 1\}$ and $m \in \{0, -1, -2, \ldots, -(m_{max} - 1)\}$ index into the $3 \times m_{max}$ grid formed by the planting pattern. The term D_0 is the offset of the central row; by definition $D_0 = 0$. It should be noted that the plant centres are assumed to be on the ground plane ($z_w = 0$). It is stressed that the model describes the grid on which individual plants should lie, rather than the actual location of the individual plants.

The origin of the world axes (x_w, y_w, z_w) is not fixed on the ground plane, but is local to the vehicle (and therefore moves along with the vehicle); as stated above the y_w axis is always coincident with the middle crop row, whilst the x_w axis passes through the point where the camera's optical axis z_c intersects the ground plane.

2.2 Observation of the Model in the Image

Having specified a model for the planting pattern, attention must now be turned to how this model will appear in the image. Owing to the angle at which the camera is mounted on the vehicle, features on the ground plane are viewed on the image plane under a perspective projection. Marchant and Brivot [8] arrive at the following expressions for the 2D image plane co-ordinates (x_u, y_u) of 3D world points $(x_w, y_w, 0)$ (t_y, t_z and ϕ are explained below):

$$x_u = f \frac{(x_w \cos\Psi + y_w \sin\Psi + t_x)}{x_w \sin\Psi \sin\phi - y_w \cos\Psi \sin\phi + t_z} \tag{3}$$

$$y_u = f \frac{(-x_w \sin\Psi \cos\phi + y_w \cos\Psi \cos\phi + t_y)}{x_w \sin\Psi \sin\phi - y_w \cos\Psi \sin\phi + t_z} \tag{4}$$

Equations 3 and 4 are derived from those given by Tsai [12], with the added assumption that the camera does not pitch or roll (i.e that the camera axes (x_c, y_c, z_c) of figure 1 do not rotate about the world axes x_w or y_w). This assumption has been made on the basis of observation of long image sequences and also because the vehicle is running on well tended tilled fields. The angle ϕ is that of the camera's optical axis (z_c) to the world z_w axis, and Ψ is that shown in figure 1. The quantity t_z is distance along the optic axis between the camera optical centre and the intersection with the ground plane, and t_y gives the offset (in camera co-ordinates) of the point where the optical axis intersects the world x_w axis and, as in [8], this can be set to zero. One further assumption is that the angle Ψ is small enough for the approximations $\cos\Psi \approx 1$ and $\sin\Psi \approx \Psi$ to hold.

From the above, it is possible to generate image pixel co-ordinates (x_f, y_f) for each plant centre (m, n) by combining equations 1 and 2 with 3 and 4, and inserting a suitable estimate of the parameters t_x, Y and Ψ

$$x_f(x, Y, \Psi, m, n) = \frac{f}{dx} \frac{nr + \Psi(ml + Y + D_n) + t_x}{nr\Psi \sin\phi - (ml + Y + D_n)\sin\phi + t_z} + C_x \tag{5}$$

$$y_f(x, Y, \Psi, m, n) = \frac{f}{dy} \frac{(ml + Y + D_n - \Psi nr)\cos\phi}{nr\Psi \sin\phi - (ml + Y + D_n)\sin\phi + t_z} + C_y \tag{6}$$

The values of dx and dy give the horizontal and vertical side length of the camera pixels respectively, and (C_x, C_y) is the co-ordinate (in pixels) of the centre of the imaging surface. In practice it has been found that the maximum number of plants seen in a single image is 15, so $m_{max} = 5$ generates a suitable number of predictions (if there are less than 15 plants in the image, then some predictions will lie outside the bounds of the image; such predictions are ignored).

3 Tracking the Pattern

The Kalman filter [5] is used to provide a means of tracking the plant model through the image sequence by *predicting* the crop structure position (and hence the individual plant positions) and using observations of plants taken from the image to *correct* this prediction. The filter estimates not only the state of a system $\mathbf{x}$, but also provides a covariance for the estimates, $\mathbf{P}$.

3.1 The System Model

The prediction is made by means of a (linear) state transition model which describes the evolution of the state $\mathbf{x} = [t_x, Y, \Psi]$ as the vehicle moves between between image k and $k + 1$:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{U}(k) + \mathbf{n}(k), \tag{7}$$

where $\mathbf{A}$ is the 3×3 state transition matrix and $\mathbf{U}$ the 3×1 control input, i.e. the vehicle motion between image k and $k+1$. By treating $\mathbf{U}$ as an external variable, rather than a state to be estimated, the kinematic model of the vehicle is hidden from the algorithm (therefore providing an estimate that is independent of a kinematic model which will be violated in certain circumstances, e.g. when the vehicle's wheels slip); the problem of integrating the vision system Kalman filter given here with the vehicle control system filter (where the vehicle kinematics are used to estimate $\mathbf{U}$) will be addressed in the future. There is also an additive zero-mean Gaussian noise term $\mathbf{n}$ which has (3×3) covariance matrix $\mathbf{Q}(k)$, and quantifies the uncertainty in the vehicle motion estimates. Because the planting pattern is assumed to be of fixed size and shape, $\mathbf{A}$ is taken to be the identity matrix.

The observation of the crop planting model in the image has already been discussed in section 2.2, and this can also be more formally stated in state-space terms:

$$\mathbf{z}(k, m, n) = \mathbf{h}[k, \mathbf{x}(k), m, n, \mathbf{w}(k)] = \begin{bmatrix} x_f(\mathbf{x}(k), m, n) \\ y_f(\mathbf{x}(k), m, n) \end{bmatrix} + \mathbf{w}(k) \tag{8}$$

The vector $\mathbf{w}(k)$ is zero-mean Gaussian noise described by (2×2) covariance matrix $\mathbf{R}(k, m, n)$. Here, the matrix $\mathbf{R}$ reflects the fact that the cauliflowers are not planted on a perfect grid, but deviate from the ideal positions predicted by the model. Note that the observation model $\mathbf{h}$ is both a function of the state $\mathbf{x}(k)$, and m and n, the integer variables which index the crop grid – so one state estimate generates $3 \times m_{max}$ observations. The observation model is also non-linear, so an extended Kalman filter [1] is required.

3.2 The Extended Kalman Filter

The extended Kalman filter allows a linearisation of the observation model about the state prediction point, and although it offers a sub-optimal solution to the estimation problem, and can converge to local minima (and therefore lose track) if poorly initialised, it offers a practical solution to the non-linear tracking problem. The filter predict-correct cycle is outlined here, with initialisation of the state prediction $\hat{\mathbf{x}}$ and its covariance $\mathbf{P}$ covered in section 4.

Owing to the fact that each state prediction produces several observations which must be incorporated into the filter to yield the corrected estimate, the predict-correct cycle takes a slightly unusual form. The "prior" prediction for each image is given by equations 9 and 10.

$$\hat{\mathbf{x}}(k+1|k) = \hat{\mathbf{x}}(k|k) + \mathbf{U}(k) \tag{9}$$

$$\mathbf{P}(k+1|k) = \mathbf{P}(k|k) + \mathbf{Q}(k) \tag{10}$$

From this single state prediction, equation 8 shows that a set of predicted observations is produced

$$\hat{\mathbf{z}}(k+1|k, m, n) = \begin{bmatrix} x_f(\hat{\mathbf{x}}(k+1|k), m, n) \\ y_f(\hat{\mathbf{x}}(k+1|k), m, n) \end{bmatrix} \tag{11}$$

Each of these predicted observations $\hat{\mathbf{z}}(k+1|k, m, n)$ must be matched to an observed image feature $\mathbf{z}(k+1, m, n)$ and incorporated into the state estimate. Feature matching is achieved using the nearest-neighbour data association procedure [9], with each associated feature being validated prior to incorporation (see below). The incorporation itself is performed using a batch update method as used in [6] (originally appearing in [13]) as opposed to the more traditional recursive estimation procedure. The batch update is preferred because a single state update leads to the prediction of several $(3 \times m_{max})$ feature locations, and when a validation procedure is used, the order of incorporation would become important if a recursive update scheme were in place – if a "poor" feature-prediction pair were incorporated first, it could bias successive predictions in such a way that "good" features would fail the validation test, leading to inaccurate tracking performance.

Armed with the predictions from equation 11, the associated image features (see section 3.3) $\mathbf{z}(k+1, m, n)$ may each be subjected to a validation test. This validation procedure allows a plant to be missing from the grid structure (without validation, if a plant were missing then the nearest-neighbour algorithm would associate a neighbouring plant with the prediction). If

$$\mathbf{v}(k+1,m,n) = \mathbf{z}(k+1,m,n) - \hat{\mathbf{z}}(k+1|k,m,n) \tag{12}$$

is the *innovation*, then feature $\mathbf{z}(k+1,m,n)$ is valid if

$$\mathbf{v}^T(k+1,m,n)\mathbf{S}^{-1}(k+1,m,n)\mathbf{v}(k+1,m,n) \leq \chi^2 \tag{13}$$

where χ^2 is a chi-square figure of merit, corresponding to a desired confidence level, and $\mathbf{S}(k+1,m,n)$ is the innovation covariance for observation (m,n), given by

$$\mathbf{S}(k+1,m,n) = \mathbf{h_x}(k+1,m,n)\mathbf{P}(k+1|k)\mathbf{h_x}^T(k+1,m,n) + \mathbf{R}(k+1,m,n). \tag{14}$$

The Jacobian $\mathbf{h_x}(k+1,m,n)$ is that of the observation equations 8, which linearises (to the first order) the observation equation about the prediction point, thus enabling a projection of the state estimate covariance $\mathbf{P}(k+1|k)$ into the image plane. Its definition is

$$\mathbf{h_x}(k+1,m,n) = \begin{bmatrix} \left.\frac{\partial x_f}{\partial t_x}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} & \left.\frac{\partial x_f}{\partial Y}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} & \left.\frac{\partial x_f}{\partial \Psi}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} \\ \left.\frac{\partial y_f}{\partial t_x}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} & \left.\frac{\partial y_f}{\partial Y}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} & \left.\frac{\partial y_f}{\partial \Psi}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} \end{bmatrix}. \tag{15}$$

Each observation passing the validation test (equation 13) is then used in the batch update of the state estimate and state covariance matrix. For each valid observation, the three quantities $\mathbf{h_x}(k+1,m,n)$, $\mathbf{R}(k+1,m,n)$ and $\mathbf{v}(k+1,m,n)$ are stored. A batch innovation vector $\mathbf{v}(k+1)$ is constructed by stacking each of the validated innovations $\mathbf{v}(k+1,m,n)$, and similarly a batch observation Jacobian $\mathbf{h_x}(k+1)$ is created by stacking the $\mathbf{h_x}(k+1,m,n)$ corresponding to the valid observations. Finally, a batch observation matrix $\mathbf{R}(k+1)$ is constructed which has block diagonal form, where the matrices on the diagonal are the validated $\mathbf{R}(k+1,m,n)$. The standard Kalman filter equations are then applied to these batch quantities to update the state estimate and state covariance matrix:

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\mathbf{v}(k+1) \tag{16}$$

$$\mathbf{P}(k+1|k+1) = \mathbf{P}(k+1|k) - \mathbf{W}(k+1)\mathbf{S}(k+1)\mathbf{W}^T(k+1), \tag{17}$$

with the covariance matrix of the stacked innovation

$$\mathbf{S}(k+1) = \mathbf{h_x}(k+1)\mathbf{P}(k+1|k)\mathbf{h_x}^T(k+1) + \mathbf{R}(k+1) \tag{18}$$

and Kalman gain

$$\mathbf{W}(k+1) = \mathbf{P}(k+1|k)\mathbf{h_x}^T(k+1)\mathbf{S}^{-1}(k+1). \tag{19}$$

The off-diagonal terms in the innovation covariance matrix $\mathbf{S}(k+1)$ (equation 18) describe the correlations between successive observations made from the single state prediction.

3.3 Feature Extraction, Association and Validation

Although the state estimation framework has been described, there are still some matters to be discussed, one of which is the method of extracting the observations $\mathbf{z}(k+1, m, n)$ from the images. The images are of real crop planted and raised in accordance with standard agricultural practice and are collected outdoors under natural illumination. The sequences thus contain both crop and weeds, with some plants missing from the row structure. The images are collected using a camera sensitive to near infra-red wavelengths where contrast between plant and soil matter is enhanced [8] (the camera has a visible light blocking filter). Two methods have been used to extract plant positions from the images; an automatic method described below, and manual selection of features by a human, which is used to evaluate the performance of the tracker independently of the feature extraction mechanism.

The automatic feature extraction method is that used in [8], which chain-codes areas of the infra-red image exceeding a grey-level threshold (currently, the vehicle uses dedicated hardware to perform this function). The centre of each chain code is calculated and used as a candidate feature to be matched to a prediction generated by the Kalman filter. It has been observed that weed features in the image tend to be smaller than the plants, and a simple threshold on chain code area is also imposed in order to reject the smallest weeds from the matching process.

Owing to the simplicity of the differentiation between plant and non-plant pixels (by use of a threshold) the chain-coder does not always produce perfect plant outlines; parts of leaves may be missed out because of shadows, or a plant made of separate leaves may be fractured into several objects because the regions between leaves are dark. The result of these shortcomings is that the "centres" produced by the chain coder do not necessarily correspond to the real centres of plants. Improved feature detection will feature largely in further research, although results show that this relatively crude method produces satisfactory results.

3.4 Covariance Matrices

Three covariance matrices are required to run the filter; a measure of process noise $\mathbf{Q}(k)$, which may be obtained from the Kalman filter used to estimate the vehicle motion parameters [3], an observation noise matrix $\mathbf{R}(k, m, n)$, which will be discussed here, and an initial value of the estimate covariance $\mathbf{P}(0|0)$, which will be provided in section 4.

The observation covariance $\mathbf{R}(k+1, m, n)$ quantifies the noise associated with the predicted observation $\hat{\mathbf{z}}(k+1|k)$ of plant centre (m, n) – the Kalman filter framework implicitly assumes that the plant centres are being extracted from the image, and although, as seen above, this is not the case, results on real images bear out the fact that this is an acceptable working assumption. Additional noise arises because the plant centres are not placed ideally on the grid specified by the model, but are found on the ground plane in perturbed positions which may be modelled by a two dimensional (x_w, y_w) Gaussian distribution whose mean is the ideal plant position. This covariance in the world-frame is given by the matrix $\mathbf{R_w} = \mathrm{diag}[\sigma^2_{x_w}, \sigma^2_{y_w}]$, and can be projected into the image frame using standard first order error propagation via equations 3 and 4, giving

$$\mathbf{R}(k+1, m, n) = \mathbf{F_{\hat{x}}}(k+1, m, n)\mathbf{R_w}\mathbf{F_{\hat{x}}}^T(k+1, m, n) \tag{20}$$

Where the Jacobian $\mathbf{F_{\hat{x}}}(k+1, m, n)$ is defined as

$$\mathbf{F_{\hat{x}}}(k+1, m, n) = \begin{bmatrix} \left.\frac{\partial x_u}{\partial x_w}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} & \left.\frac{\partial x_u}{\partial y_w}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} \\ \left.\frac{\partial y_u}{\partial x_w}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} & \left.\frac{\partial y_u}{\partial x_w}\right|_{\hat{\mathbf{x}}(k+1|k),m,n} \end{bmatrix} \tag{21}$$

x_w, y_w for the plant centres are defined in equations 1 and 2. As noted in the introduction, this noise source should manifest itself in the system noise model, but as yet it is unclear how this should be done; the method given here provides a working alternative.

4 Starting the Algorithm

To start the tracking process, an initial estimate of the crop pattern position $\mathbf{x}$ is acquired in two stages; the initial estimates of t_x and Ψ are obtained by a global search of the Hough space which is formed using the methods of [8]. The model position Y and row offsets D_{-1} and D_1 are then obtained from Fourier analysis of 1D image samples taken along the extracted rows. By sampling along the line of the rows in the image a profile of the grey-levels along the row is obtained. Figure 2 shows an idealised binary image with a row marked and its corresponding sample. The sampling is performed in world frame co-ordinates, which accounts for the regular spacing of the sample peaks in the figure, despite the perspective foreshortening in the image. To allow for non-ideal positioning of the row, and the fact that plants are two dimensional objects in the image rather than one dimensional, the sample analysed is constructed by taking the mean (at each sample point) of a set of 5 samples taken 5 mm apart on the ground plane around the row specified by the Hough transform method.

The procedure for obtaining offset values from these samples is as follows; form the discrete Fourier transform $F(j\omega)$ of the (1D) grey-level profile (ω is the angular spatial frequency in radians per metre) and calculate the phase θ of the coefficients corresponding to the frequency $2\pi/l$, the expected frequency of plant spacing l from the crop model. θ can be converted into spatial offset along the row using the following formula

$$\text{Offset} = \frac{\theta l}{2\pi}. \tag{22}$$

Using this method, model position Y can be calculated from the central row, and D_{-1}, D_1 found subsequently from the outer rows.

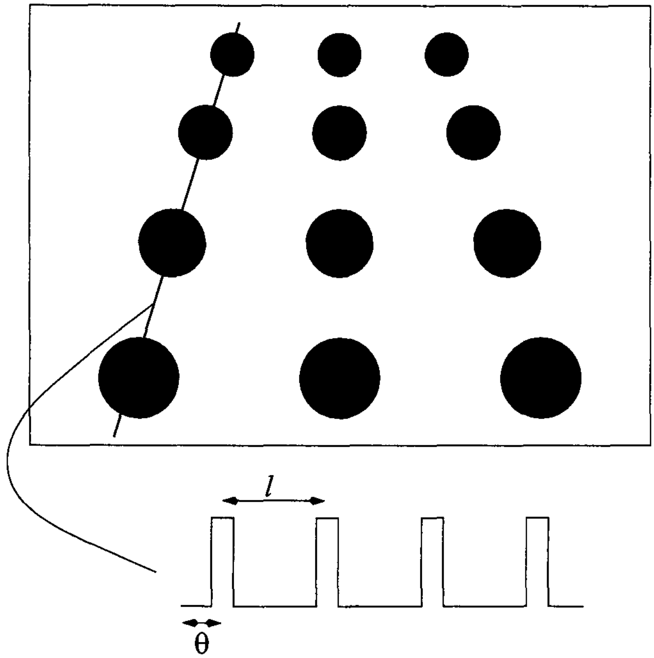

Fig. 2. Sampling along a row: the phase θ of the sample provides the offset of the planting pattern

Once initial values for t_x, Y and Ψ have been obtained, the state estimate $\hat{\mathbf{x}}(0|0)$ may be formed, leaving only the initial state covariance $\mathbf{P}(0|0)$ undetermined. From [8], an estimate of root mean square error has been obtained on the accuracy of the Hough transform algorithm, giving r.m.s error of 12.5 mm on t_x and 1^o on Ψ. As noted in [1], in the extended Kalman filter the matrix $\mathbf{P}$ is not strictly a covariance, but a measure of mean square error on the estimate $\hat{\mathbf{x}}$, so these values of offset and angular error may be used directly. To obtain a measure of mean square error for the estimate of row offset, the method of [8] was used; the row offset algorithm was applied to 40 images, and estimates of the central row offset were acquired. A human operator was then asked to align a template of plant centres with the crop in each image, the position of the template giving the offset; the initial alignment of the template with the row structure was carried out using the Hough transform method, so that only the Fourier transform part of the algorithm was under test. A scatter plot of the human *vs* automatic measurements of offset is shown in figure 3. A regression line was then fitted through this set of points, and the r.m.s. error calculated, with the resulting error being 24.5 mm. It should be noted that this method implicitly assumes that all the errors arise from the algorithm (i.e. that the human assessment is perfect), so is far from ideal. However, with a lack of veridical data,

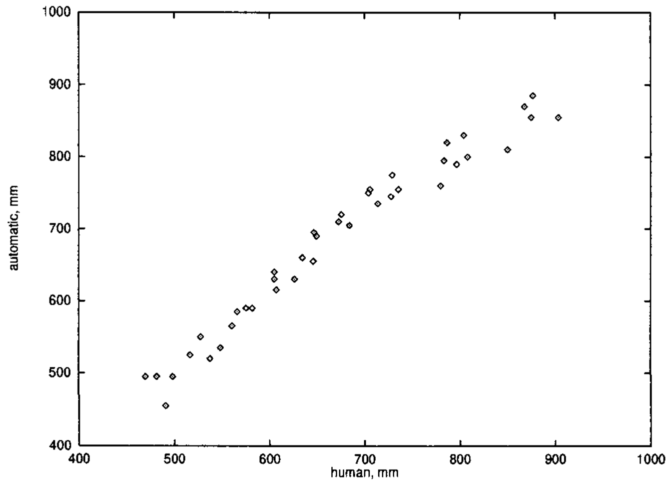

Fig. 3. Comparison of human and automatic assessment of row offset

this kind of comparison with subjective human judgement provides a pragmatic solution to the problem of estimating the initial state covariance.

Experiments in which the initial covariance was increased by up to a factor of 10, or decreased to zero showed that this only affected the initial response of the filter, with convergence to the same track after two or three images into the sequence.

5 Results and Discussion

Two sets of 20 near infra-red images have been digitised from video stock collected from the experimental vehicle during the Summer of 1997. In both sequences, the vehicle was instructed to follow the crop rows at a constant velocity. The image sequences were analysed using the following methods:

1. The fully automatic algorithm described in this paper (AUTO).
2. The Kalman filter using the human selected input features (SEMI).
3. Human assessment of the model position; a mouse-driven program has been designed to allow the user to place the crop pattern on each image in the sequence. Data from three different people has been collected (HUMAN 1–3).
4. The method of [8], which produces estimates of t_x and Ψ alone (HOUGH).

Figure 4 shows state trajectories from the first image sequence. The left-hand column plots the three human responses, whilst the right hand column shows the equivalent automatic results (note that there is no Y estimate available from the Hough transform algorithm).

As noted above, when ground truth trajectories are unavailable, as in this case, quantitative analysis of the accuracy of a tracking system is difficult to perform

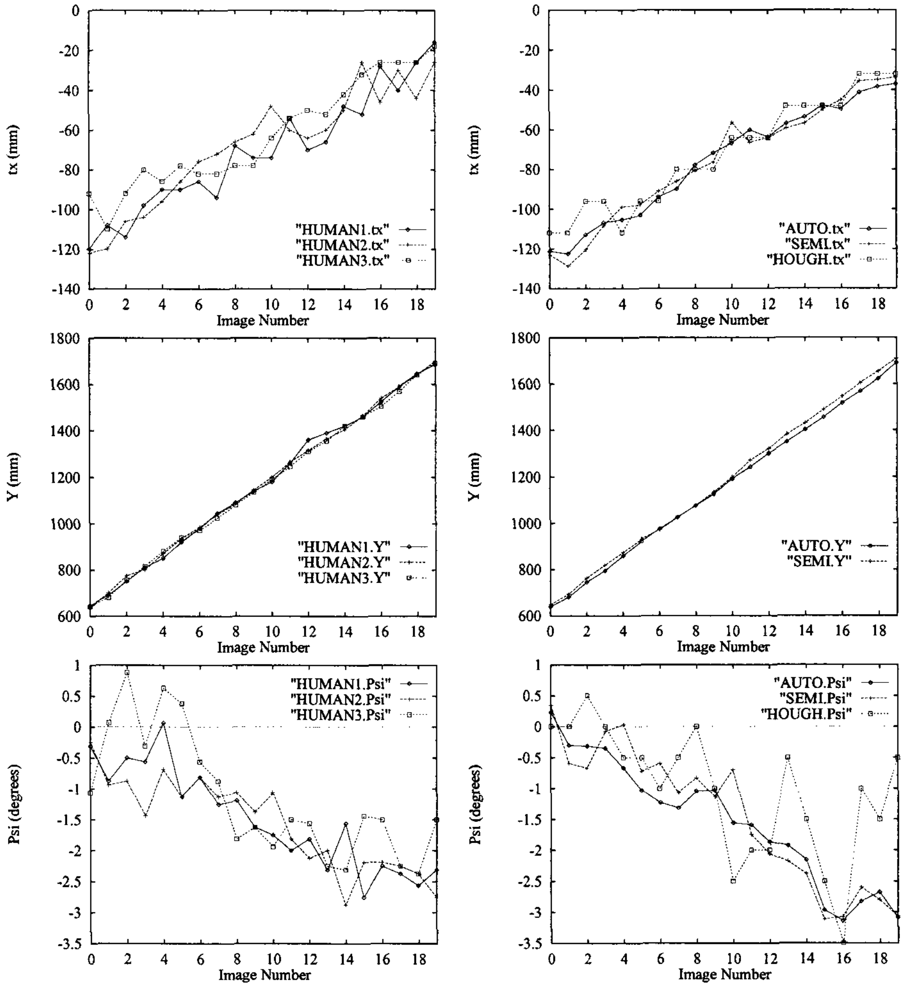

Fig. 4. Trajectories of the state variables. In the left-hand column are human assessments, and in the right the algorithm output (note that HOUGH does not produce a Y output). The negative values of t_x indicate that the vehicle was to the left of the planting pattern, but moving toward it. The Y estimates have been plotted with respect to a static global origin (as opposed to the moving co-ordinate system described in section 2) to illustrate clearly the vehicle's approximately constant velocity.

because it is not known how the errors are distributed between the automatic methods and human assessment. To provide some measure of performance, an approach has been taken which assumes errors are equally distributed between the automatic and human approach. By taking each set of results for t_x,Y and Ψ from the experiments conducted and pairing of corresponding data sets, scatter plots like figure 3 can be constructed. If a pair of algorithms agree exactly, then the points in the scatter diagram will lie on the line $x = y$. Taking this as the ideal response, a measure of how far a pair of algorithms depart from the ideal may be found by taking the root mean-square differences between feature points in the scatter plot and the nearest (in Euclidean terms) point on the line $x = y$. These values are tabulated for each of the state variables and algorithm pairs in tables 1 – 3. It should be stressed that the $x = y$ "ideal" does not mean that a pairing is correct, but that the two sets are consistent, so the larger this measure, the greater the inconsistency between them.

From the tabulated figures and perusal of the trajectory plots, three main conclusions may be drawn;

1. Human assessments are not wholly consistent. The table elements referring to the similarity between HUMAN data set pairs contain figures that are not zero; so, unsurprisingly, different people make differing assessments of the model position.
2. The various algorithm estimates are as consistent with the human results as the human results are with each other. In some cases, notably the t_x estimate of HUMAN 2, the human observation is more consistent with the automatic and semi-automatic methods than with the other humans. Importantly, the new method (AUTO) has similar r.m.s. measures of consistency to the HOUGH method, which has operated successfully on the vehicle. For the measurement of Ψ, the AUTO method is more consistent with human assessment than HOUGH; this is reflected in the plots of figure 4.
3. The fully automatic algorithm performs comparably with the semi-automatic algorithm. The similarity measure between the two is given in the table; in the case of t_x and Ψ, the differences are a fraction of the quantisation interval used in the Hough transform method, so it may be assumed that errors in the automatic method would make no operational difference, and the mean difference of 14 mm on a measurement of Y ranging from 600 – 1700 mm is also very small. It can be seen that when compared to the human assessments, the two differ even less.

6 Conclusions

A self-starting algorithm has been demonstrated which allows the extraction of crop planting patterns from a sequence of images, giving information for both vehicle navigation and plant treatment system control. The new method produces estimates of offset t_x and heading angle Ψ which compare favourably with those produced by the Hough transform algorithm, and furthermore yields both

	HUMAN 1	HUMAN 2	HUMAN 3	HOUGH	AUTO	SEMI
HUMAN 1	0	–	–	–	–	–
HUMAN 2	8.02	0	–	–	–	–
HUMAN 3	7.54	7.96	0	–	–	–
HOUGH	8.20	7.70	7.85	0	–	–
AUTO	6.63	6.41	8.89	5.06	0	–
SEMI	6.76	6.17	8.85	5.86	2.82	0

Table 1. Root-mean square differences on the t_x estimate, mm. '-' indicates that the value can be found in the lower half of table.

	HUMAN 1	HUMAN 2	HUMAN 3	HOUGH	AUTO	SEMI
HUMAN 1	0	–	–	n/a	–	–
HUMAN 2	14.20	0	–	n/a	–	–
HUMAN 3	13.72	13.84	0	n/a	–	–
HOUGH	n/a	n/a	n/a	n/a	n/a	n/a
AUTO	13.18	17.19	10.16	n/a	0	–
SEMI	12.55	10.82	12.50	n/a	14.11	0

Table 2. Root-mean square differences on the Y estimate, mm. 'n/a' indicates Y is not estimated by algorithm HOUGH.

	HUMAN 1	HUMAN 2	HUMAN 3	HOUGH	AUTO	SEMI
HUMAN 1	0	–	–	–	–	–
HUMAN 2	0.408	0	–	–	–	–
HUMAN 3	0.510	0.540	0	–	–	–
HOUGH	0.750	0.808	0.698	0	–	–
AUTO	0.363	0.365	0.510	0.666	0	–
SEMI	0.426	0.323	0.532	0.742	0.253	0

Table 3. Root-mean square differences on the Ψ estimate, degrees. The figures indicate that the set least consistent with the other data is from algorithm HOUGH, and this is reflected in the plots of figure 4.

an estimate of the Y measurement which is used to locate individual plants for treatment, and a measure of uncertainty $\mathbf{P}$ which will be utilised by the control system Kalman filter in the fusion of vision information with odometric measurements.

It is hoped to implement the algorithm to run in real time (which should be feasible due to the computational simplicity of the method) and to test it on longer image sequences. Incorporation of this method into the vehicle control system will also be addressed. A method of on-line model parameter adjustment is to be added in order to correct for any calibration errors, and to compensate for any gradual change in crop spacing which may occur during planting. It is also believed that more sophisticated feature extraction techniques may improve performance, particularly robustness to the changing lighting conditions often experienced in outdoor environments.

Acknowledgements

Thanks to Robin Tillett and Dickson Chan for their patience in producing the HUMAN 2 and HUMAN 3 data sets respectively. This work was funded by the BBSRC.

References

1. Y Bar-Shalom and T Fortmann. *Tracking and Data Association.* Academic Press, New York, 1988.
2. T Hague, J A Marchant, and N D Tillett. Autonomous robot navigation for precision horticulture. In *IEEE International Conference on Robotics and Automation.* Albuquerque, 1997.
3. T Hague and N D Tillett. Navigation and control of an autonomous horticultural robot. *Mechatronics*, 6(2):165–180, 1996.
4. C Harris. Tracking with rigid models. In A Blake and A Yuille, editors, *Active Vision*, chapter 4. MIT Press, 1992.
5. R E Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME Journal of Basic Engineering*, 1960.
6. J J Leonard and H F Durrant-Whyte. Mobile robot localization by tracking geometric beacons. *IEEE Trans. Robotics and Automation*, 7(3):376–382, June 1991.
7. J A Marchant. Tracking of row structure in three crops using image analysis. *Computers and electronics in agriculture*, 15(9):161–179, 1996.
8. J A Marchant and R Brivot. Real time tracking of plant rows using a hough transform. *Real Time Imaging*, 1:363–371, 1995.
9. B Rao. Data association methods for tracking systems. In A Blake and A Yuille, editors, *Active Vision*, chapter 6. MIT Press, 1992.
10. D Reynard, A Wildenberg, A Blake, and J A Marchant. Learning dynamics of complex motions from image sequences. In B Buxton and R Cipolla, editors, *Computer Vision – ECCV '96*, Lecture Notes in Computer Science. Springer, April 1996.
11. J M Sanchiz, F Pla, J A Marchant, and R Brivot. Structure from motion techniques applied to crop field mapping. *Image and Vision Computing*, 14:353–363, 1996.
12. R Y Tsai. An efficient and accurate camera calibration technique for 3d machine vision. In *Proc. IEEE Computer Soc. Conf. Comp Vis Patt Recog*, Miami Beach, June 1986.
13. D Willner, C B Chang, and K P Dunn. Kalman filter algorithms for a multi-sensor system. In *Proc. IEEE Int. Conf. Decision and Control*, pages 570–574, 1976.