ELSEVIER

Image and Vision Computing 15 (1997) 465–473

image AND vision COMPUTING

Application note

Building perspective models to guide a row crop navigation vehicle

F. Pla[a], J.M. Sanchiz[a,*], J.A. Marchant[b], R. Brivot[b]

[a]*Department of Computer Science, Campus Penyeta Roja, University Jaume I, 12071 Castello, Spain*
[b]*Silsoe Research Institute, Wrest Park, Silsoe, Beds MK45 4HS, UK*

Received 23 January 1995; accepted 12 November 1996

Abstract

To fully automate some in-field agricultural tasks, an autonomous vehicle has to be guided along a crop field. In this work, we describe how a vision system extracts the guidance information for vehicle navigation in a crop row. The method is based on extraction of the scene structure from perspective information that a set of coplanar parallel lines, representing the crop rows, generate in the image. To solve some drawbacks caused by these natural environments, a method to extract the row lines is described, and an approach to calculate the vanishing point proposed. A technique to recover missing information is also introduced. The method has been applied to long sequences of images taken by a moving vehicle in crop row scenes.

Keywords: Guidance; Perspective; Vanishing point; Row crops

1. Introduction

Vehicle navigation forms an essential part of any fully automated system that is to perform some agricultural tasks in a natural environment. The work presented here is part of a visual sensing system to guide an autonomous crop protection vehicle. In particular, we are interested in row crops, i.e. crops where plants are distributed in a regular geometrical structure, a set of parallel lines lying in the ground plane. The vision system has to extract information about this geometrical structure to orientate the vehicle, and to determine the vehicle's position with respect to the crop structure at every moment.

Vision-based guidance for vehicle navigation has largely been of interest in road following and obstacle avoidance. In road following, the vehicle has to be driven along a road in an outdoor environment [1–5], and in indoor environments within a room or in corridors [6–8]. For longer distance navigation, a mobile robot will also have to understand the route it travels, representing and memorizing its environment while moving around looking for landmarks by which to orientate itself, either using a bird's eye map [4,9] or a series of route views stored previously [10].

* Corresponding author.

PII S0262-8856(96)01147-X

In our case, we have a well defined model of the terrain map, and the problem consists of how to extract information to understand the route the vehicle travels, finding the present location of the vehicle with respect to the terrain map, and planning the trajectory to follow. In the case of row crops, plants are distributed along lines equally separated, forming a parallel geometrical structure, which represents the terrain map. The projection of this structure in the image plane converges in a vanishing point.

The work presented here differs from previous guidance work because the motivation or application domain leads us to consider other techniques to extract the image features from which guidance information is drawn. Unlike man-made scenes (road, buildings, and so on), objects from scenes in natural environments do not have regular and defined shapes; they have a high texture component and object borders with random shapes.

Instead of working out the vanishing point after the scene structure has been found, as in road following approaches, we use the vanishing point to build the scene structure. In addition, the vanishing point, together with some prior knowledge about the scene, is used to infer from perspective geometry concepts possible missing information in the scene model. Since most of the method described here is based on the vanishing point of the perspective structure, an accurate method to calculate the vanishing point is needed. A method to work out the vanishing point is presented which overcomes the limited accuracy of Hough transform-like techniques.

2. Feature extraction

Most work on vision systems to guide vehicles is applied to man-made environments [1–10], and one of the most common techniques used in preprocessing is to apply an edge detector, looking for straight lines [1,6]. Unlike other outdoor environments containing man-made objects like buildings, corridors and roads, scenes from natural objects do not contain highly defined straight borders and planar surfaces.

2.1. Image segmentation

Many road following approaches use image segmentation techniques to extract the basic features with which to guide the vehicle, like texture segmentation [1,11] or grey-level segmentation [1,12]. Nevertheless, most segmentation-based approaches use colour segmentation [3–5] as a method which can deal with a greater variety of situations, mainly in outdoor scenes. Moreover, the use of colour segmentation becomes a complex problem in outdoor environments with the presence of shadows, illumination changes and a continuous change in weather conditions [4].

To avoid the above-mentioned illumination problems, a colour segmentation method based on properties from the dichromatic reflection model is used, to achieve colour recognition independent from illumination changes [13]. The scene illuminant colour and samples from each colour class have to be provided to train the colour classification parameters. After the colour classifier has been trained using samples from one image of a sequence, every pixel in the sequence of images is classified into one of the colour classes, producing a labelled image, where regions labelled as plant colour are identified as plants which form the scene structure.

2.2. Extracting region skeletons

Because regions resulting from the colour segmentation do not have straight borders, we use the skeleton of each defined region as a feature to work out the lines which define the crop. Thus, a thinning algorithm [14] is used on the regions of interest. The resulting skeletons of each region can be used as curves which define the underlying structure of the crop, and to extract the straight lines where the plants and soil rows lie.

Skeletons resulting from thinned regions are usually fairly noisy, with a lot of contour segments emerging from the real spine of the plants arranged as lines. These extra segments in the skeleton can produce many straight line segments, which increase the computational cost to search for the vanishing point, and they might mislead the vanishing point detection procedure. To avoid this drawback, an algorithm to cut off noisy skeleton branches has been designed.

The skeleton pruning algorithm uses a criterion based on two parameters: the type of segments in which the skeleton segments are classified; and the length of the skeleton segments. Segments in the skeleton are defined as chains of connected contour points between two control points. Control points are points where two or more contour segments join, or the end of any open contour.

To remove the noisy parts of the skeleton after thinning, the algorithm shown in Fig. 1 is used. In this algorithm, a *hair* is a segment bound by an end point and a junction point, and a *double branch* is represented by two skeleton segments that are bound by the same junction points. An example of the result of this algorithm is shown in Fig. 7, where we can see how the resulting skeleton represents the spine of the original one (Fig. 6), avoiding some extra skeleton segments.

2.3. Straight line extraction

After skeletons have been pruned, an algorithm to extract lines from the skeleton curves is used. In this case, we have used an approach based on an algorithm which, after segmenting the skeleton curves into constant curvature segments, group these segments in such a way

```
Algorithm: Skeleton prunning

Parameter: D_min ; minimum length of a skeleton segment.

do
    continue := false;
    For each hair in the skeleton
        If it joins another hair
            Choose either of them;
            Cut the other hair off;
        else if Length(hair) < D_min
            Cut the hair off;
    For each double branch
        take the larger branch;
        If Length(branch) < D_min
            Cut the branch off;
            continue := true;
while continue = true;
```

Fig. 1. The skeleton pruning algorithm.

Algorithm: Estimate missing lines

1. Order the crop lines found in the image plane counterclockwise.
2. Reject the lines which are almost horizontal, since they can introduce noise.
3. Compute the separation between consecutive lines up to a scale factor, fixing h to a constant.
4. Compute the mean μ and standard deviation σ of these separations.
5. Compute the relative error as μ/σ.
6. If the relative error is larger than a maximum allowed error, delete the separations that differ from the mean more than the standard deviation. If no separations have been deleted, go to step 7, otherwise go to step 4.
7. Thus, μ is supposed to be the separation distance l between consecutive lines along the line $y'=h$. Therefore, for each separation l, the factor $k=d/\mu$ is computed. If k is rounded up to an integer, k-1 denotes the number of lines missed between the lines separated a distance l.
8. To insert the missed lines, use the parameters m and n from the lines found, to calculate the paramters m' and n'' of contiguous missed lines as explained in the text.

Fig. 2. The algorithm to estimate missing lines in the row crop structure.

that all segments in a group represent a straight line. This approach gives a small number of primitive (straight lines) to calculate the vanishing point. In addition, apart from the straight lines, the algorithm also provides knowledge about the segments (and thus the contour points) that define each line found, i.e. each line is always associated with a set of points which support the line. This information is used in the decision about the identification of the scene structure's vanishing point. The algorithm used is based on that proposed by Pla [15], but instead of grouping contour segments that form part of a circumference, it groups contour segments that form part of the same straight line.

3. Vanishing point detection

After all possible lines have been found, a vanishing point detection algorithm is performed to bring out the point where all lines which define the crop join. Since we are working with a moving vehicle taking sequences of images, we have to differentiate between vanishing point detection at the boot-strap and feed-forward phases. In the feed forward phase, the vanishing point is detected using previous information about the vanishing point found in the previous image of the sequence, performing a sort of tracking on the vanishing point.

3.1. Boot strap

Among the algorithms used to calculate vanishing points in the literature, there is one which has a performance of order N (where N is the number of the initial set of lines) [16], which can only be applied to what the authors call the 'legoland world'. Most of the vanishing point detection algorithms [17,18] are Hough transform-like methods, which have a performance of order N^2, approximately. Other approaches use probabilistic methods, which have a combinatorial cost [19].

In this work, we propose an algorithm which has performance of order N^2 in the average case. The algorithm uses as input the coordinates of all possible points resulting from the intersection of line pairs. Given the set of these intersecting points, an algorithm based on a single threshold sequential clustering [20] searches for clusters of points. These clusters of points represent potential points of intersection between lines. A criterion to choose one of the clusters is used to identify the cluster which represents the vanishing point of the row crop structure.

The criterion used to choose a cluster C_p from the resulting list of clusters C is based on a measure of the support each cluster receives from each point of such a cluster. Recall that each point of the cluster has two associated lines. Thus, each cluster C_i has associated a set of lines $L_i = \{l_1, l_2, \ldots l_{ni}\}$. At the same time, each line l_k is supported from a number of skeleton points $nsp(l_k)$ from which the line has been found, as described in the previous section. Therefore, we associate a score s_i with each cluster C_i, $i = l, \ldots, M$; consisting of

$$s_i = \sum_{k=l}^{ni} nsp(l_k)$$

Assuming that most contour points in the image, used to extract the lines, belong to the crop structure of parallel lines, the cluster which represents the vanishing point of the structure will be the mean point r_p of the cluster C_p supported by the maximum number of contour points, that is, $s_p = \max\{s_i\},\ i = l, \ldots M$.

Although this algorithm has cost N^2, the same order as the Hough transform-like methods, it has some advantages with respect to these methods. One advantage is that there is no need of a Hough space, therefore memory requirements only depend upon the list of crossing points. In addition, because of the clustering algorithm, the problems of finding peaks in the Hough space are overcome, even in the case where the number of initial lines is small. The main advantage of the algorithm is the accuracy in the calculation of the vanishing point, since it is not restricted by the size of the cells in the Hough space.

3.2. Feed forward

In subsequent images, information about the previously calculated vanishing point is used to work out the vanishing point in the present image, performing a tracking on this point. To track the vanishing point in this way, motion between frames is supposed to be small, therefore the relative displacement of the vanishing point from one image to another is supposed to be small. After applying the same line extraction process to any image of the sequence, the calculation of the vanishing point in any image of the feed-forward phase consists of:

1. Define a circular region around the coordinates of the vanishing point of the previous image.
2. Select the set of lines in the current image which cross the defined circle.
3. Compute the present vanishing point from this set of lines, as the point which has a minimal distance to this set of lines.

This problem has a performance of order N, N being the number of extracted lines in the image, and it reduces the computational time spent with respect to the bootstrap phase.

4. Extracting guidance information

Once the vanishing point r_p of the crop structure has been found, its corresponding cluster C_p has associated the set of lines L_p which meet at this vanishing point. Therefore, this set of lines represents the lines of crop where the plants or the soil rows lie. Having identified the lines L_p that represent the structure of the crop, we can bring out the relative direction between the direction in which the vehicle is moving at the moment, and the direction we want the vehicle to take. The vehicle has to move along the row lines of the crop, so a measure of reference is the relative direction between the line in the image which represents the forward direction of the vehicle, and the direction of the straight line of the soil row where the vehicle is moving at present.

From the model found, the row we want to follow is chosen as the row nearest to the soil row which was taken as the path to follow in the previous image. Given the set of lines L_p, which define the row crop structure, the nearest line to the line chosen in the previous frame l'_p, is the one at which the intersecting point at the bottom of the image is nearer to the intersecting point of the line l'_p at the bottom of the image.

4.1. Robust recovery of the crop structure

In case the row line used as the reference path is missed during the line extraction process, an algorithm is used to allow the vision system to estimate the position of this line, or others, from the lines and the vanishing point found. The algorithm is based on some perspective properties derived from the geometrical structure of the scene we are dealing with.

Let us assume that the swing angle ψ of the camera with respect to the ground plane is approximately constant and null, since the vehicle movement is approximately carried out on the ground plane (Fig. 3). Let $(x, y, z) = ((h - n)/m, h, 0) + t(x'_{vp}/f,\ y'_{vp}/f,\ 1)$ be the row line in 3D which crosses the point (x_0, y_0, z_0), and let $y = m x' + n$ be its projection line in the image plane. Its contiguous line separated a distance l along the straight line $x' = h$ (Fig. 3) will have as the equation $(x, y, z) = ((h - n)/m + l,\ h,\ 0) + t\ (x'_{vp}/f, y'_{vp}/f, 1)$. The projection of this line in the image plane, $y' = m' x' + n'$, must cross the vanishing point (x'_{vp}, y'_{vp}) and the point $((h - n)/m + l, h)$. Then m' and n' can be calculated by forcing their values to satisfy the equation $y' = m'\ x' + n'$ at the points $(x'_{vp},\ y'_{vp})$ and $((h - n)/m + l,\ h)$, that is,

$$m' = (m h - m y_{vp})/(h - n + m l - x_{vp} m)$$

$$n' = y'_{vp} - m' x'_{vp}$$

Distance l can be computed from the distance between the crop rows d, which is also known, as well as the focal length of the camera f. However, to make this calculation independent from any measure supplied *a priori*, and since we are only interested in calculating the parameters of the image plane, we can fix h to a given constant, and therefore work out the value of distance l up to a scale factor. Thus, the algorithm to recover missing lines of the crop structure is represented in Fig. 2.

5. Results and discussion

A colour camera was mounted on a moving vehicle

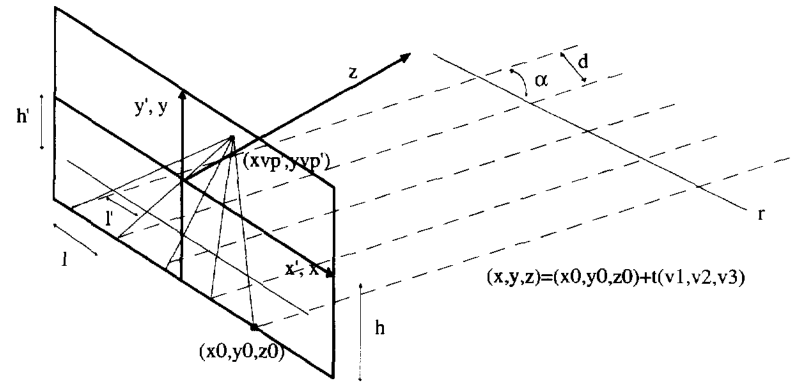

Fig. 3. Projection in the image plane of a set of parallel lines equally separated, and lying in the same ground plane. Swing angle Ψ is considered null.

commanded manually, in order to simulate a real moving vehicle, and to take real image sequences in the field. The camera is assumed to undergo the same movement as the steering wheel of the vehicle, with the optical center of the camera on the steering axis, and the image plane perpendicular to the forward direction of the steering wheel. The speed of the vehicle was about 0.5 meters/second, and the image capture rate was of 4.17 images/second. Image capture was done with a colour frame grabber on a transputer system under natural illumination conditions.

Fig. 4 shows an image of a sequence, and Fig. 5 represents the colour segmented image. In this case, skeletons were extracted from soil regions. After regions in Fig. 5 have been thinned (Fig. 6), the result of the skeleton pruning algorithm is shown in Fig. 7. Note how the pruning has cut off all the noisy segments of the skeleton, maintaining the underlying structure of the row lines. Fig. 8 represents all the possible straight lines found from the pruned skeleton. Fig. 9 shows the vanishing point and the lines associated found using the algorithm described in the boot-strap phase. Angle θ between the line which represents the steering direction and the line chosen as the path to follow, and distance x_θ in the x-axis at the bottom of the image, are the parameters used to control the vehicle movement.

The vanishing point was successfully calculated during the feed-forward phase for all images of the sequences, and the line row to follow successfully chosen. To show the robust recovery of the crop structure, Fig. 10 represents the structure found in Fig. 9, but removing two lines of the crop simulating a failure in the feature extraction process. Fig. 11 shows the lines recovered from the information of other lines present in the structure using the algorithm described previously. Note that the recovered lines are practically the same as the true lines represented in Fig. 9. Therefore, this algorithm can recover temporarily missed information to maintain the movement until real information is provided.

Fig. 12 represents the skeleton extracted, after pruning, from the plant regions in Fig. 5, and the result of the vanishing point detection is shown in Fig. 13. When the information is extracted from plants, the line to be chosen as the path to follow is calculated from the middle lines between two contiguous plant lines. The decision about extracting the crop structure either from plant regions or soil regions depends upon the crop age. The more the crop grows up, the more the soil regions recede and plant regions represent the greater part of the image, therefore the thinning algorithm is more effective using soil regions.

Fig. 14 represents an original image of the same crop in Fig. 4, but some weeks later. Fig. 15 shows the colour segmentation of Fig. 14. Note how the plant regions represents the greater part of the image, and how it has become more difficult to extract the crop structure from these regions. The vanishing point found is shown in Fig. 16. The vanishing point was successfully found in all 30 images of the sequence represented by this image.

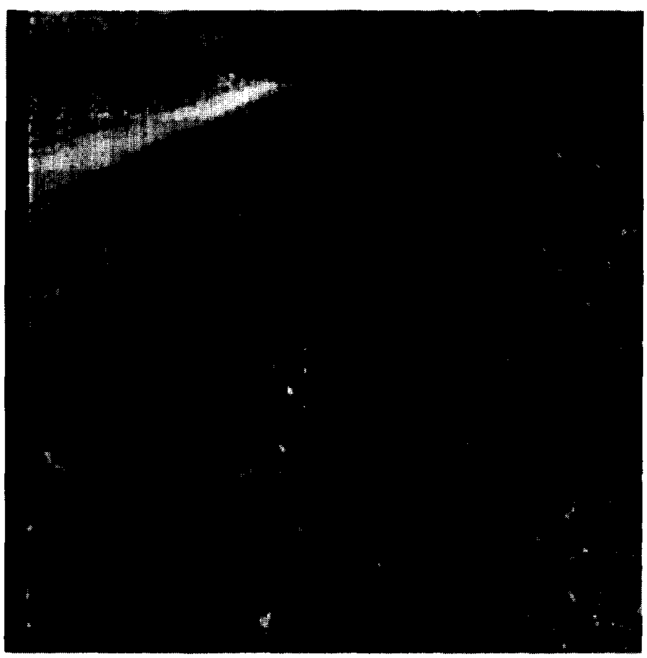

Fig. 4. Original image of a row crop scene.

Fig. 5. Colour segmented image of Fig. 4.

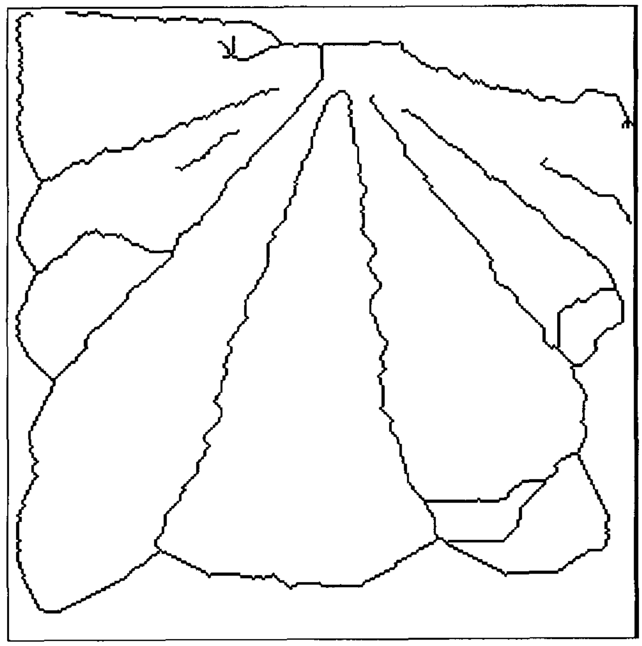

Fig. 7. Skeletons from Fig. 6 after applying the skeleton pruning algorithm.

An additional problem arises when the vehicle is reaching the end of the crop, then the perspective information disappears. One possibility is to extract the motion of the vehicle and try to maintain a straight line in the last meters of the crop line. Another possibility being considered is to look backwards with another camera, using the same type of perspective information.

Implementation of the method explained in real time is entirely feasible. Colour segmentation, after training, can be performed by input LUT operations, or using a threshold in infrared images [21]. There exist parallel thinning algorithms which can perform the thinning at video rate [22], with the appropriate hardware. Skeleton pruning and vanishing point extraction during the feed-forward phase do not have a significant computation cost.

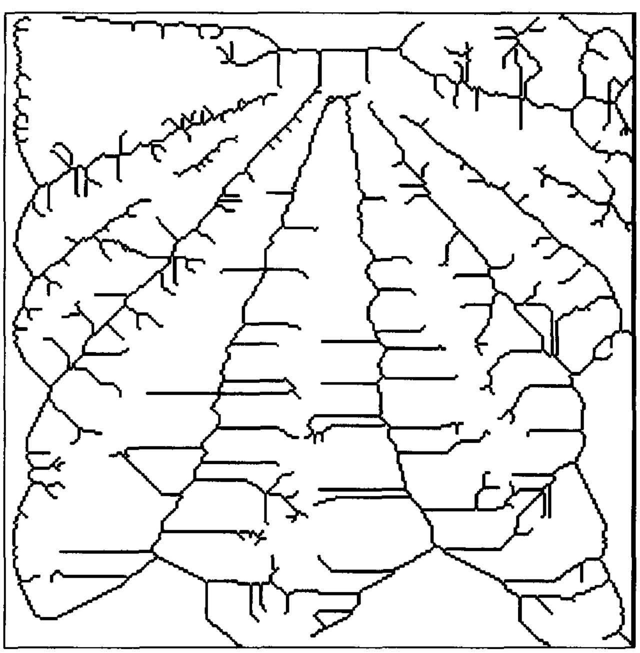

Fig. 6. Skeleton resulting from the thinning of soil regions in Fig. 5.

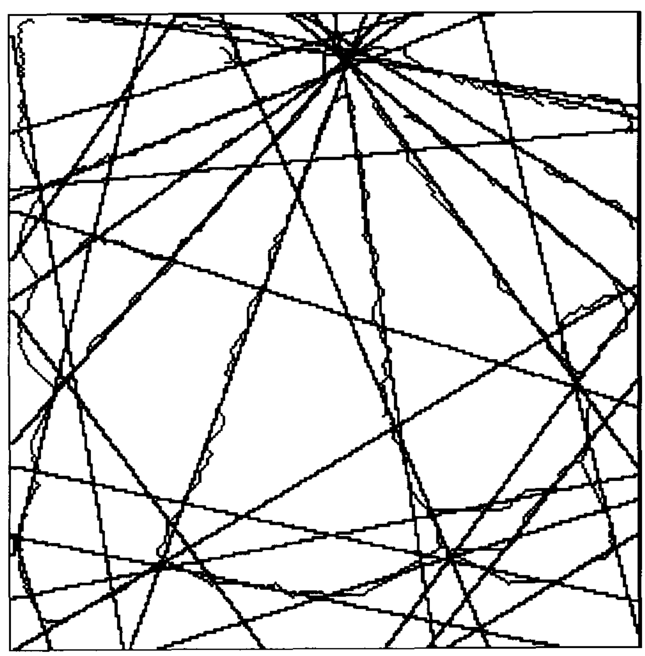

Fig. 8. Straight lines found from the skeletons in Fig. 7. Skeleton segments which support the lines are also shown.

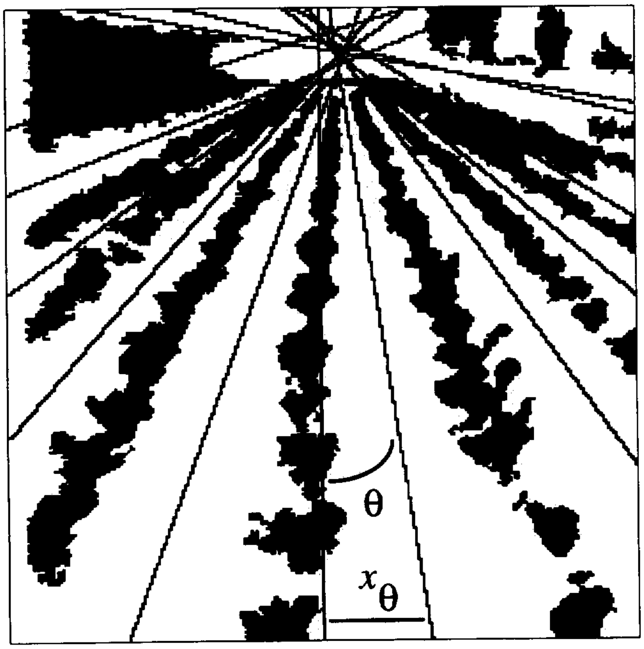

Fig. 9. Vanishing point and its associated lines found to build the perspective model of the row crop. Steering direction is also represented to show the angle between the steering direction and the line to follow, θ, and the distance at the bottom of the image between the steering line and the line to follow, x_θ.

6. Conclusions and further work

We have presented a method to extract visual information to guide a crop protection vehicle. This method is based on image analysis techniques to interpret the geometrical structure of the crop from perspective information. The main features of the method proposed are:

- a feature extraction method based on region skeletons has been described;
- a method to calculate vanishing points, overcoming some disadvantages of Hough transform-like techniques, and an approach for the feed-forward phase to reduce computational time;

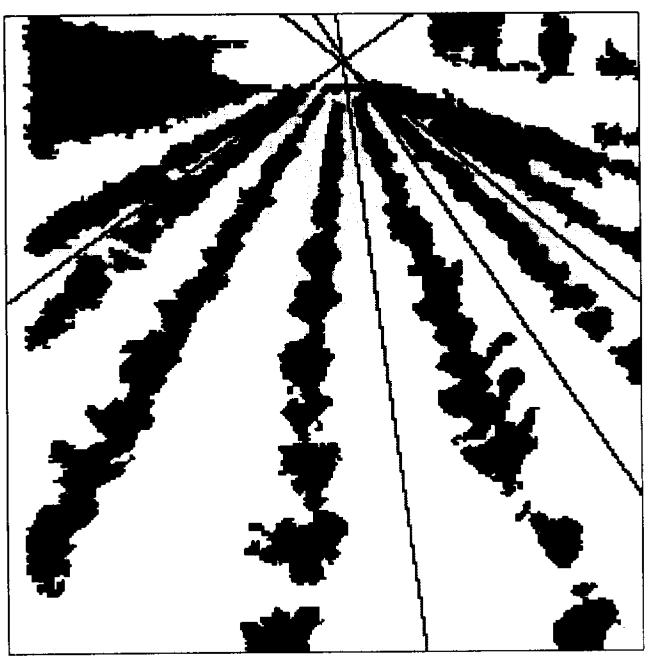

Fig. 10. The same as Fig. 9, but removing two crop lines to simulate a failure in the construction of the scene model.

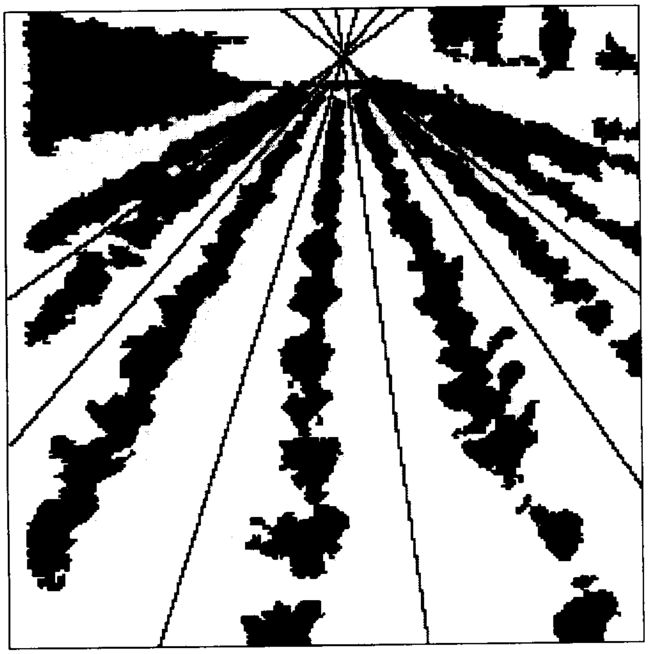

Fig. 11. Recovered lost lines from Fig. 10 using the information from the other lines found and the vanishing point.

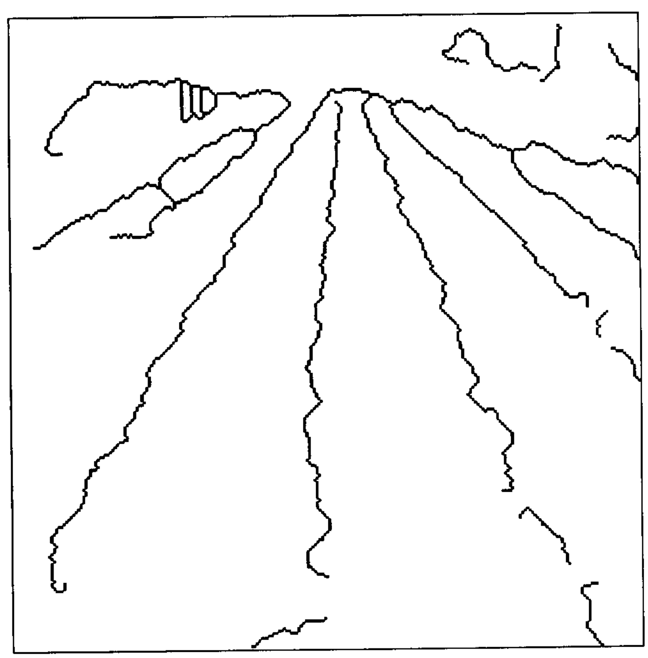

Fig. 12. Pruned skeleton from plant regions in Fig. 5.

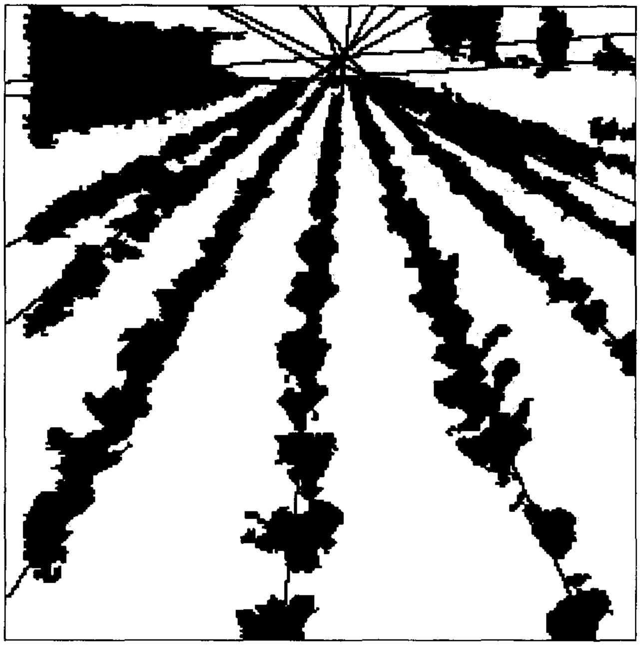

Fig. 13. Perspective model extracted from skeletons in Fig. 12.

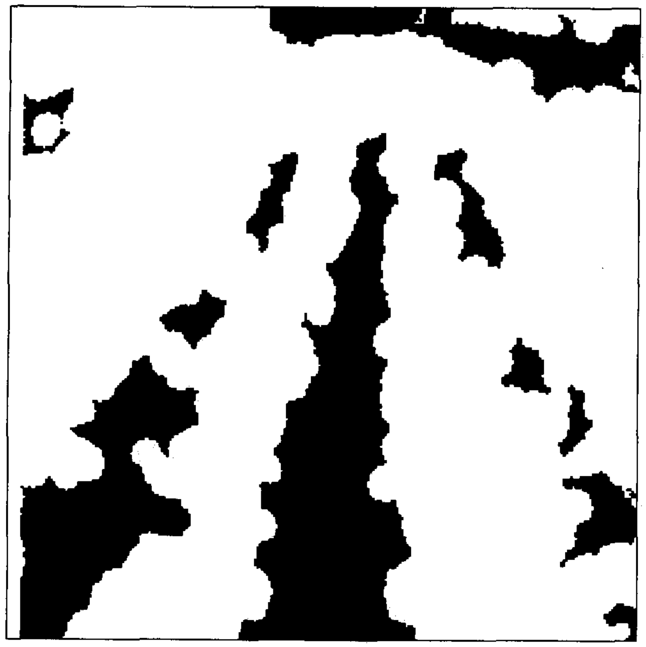

Fig. 15. Colour segmented image from Fig. 14.

- an algorithm to recover possible missing lines in the crop structure.

The experiments carried out so far show that the method has been applied successfully to long sequences of images from row crops. An additional problem arises when the vehicle is arriving at the end of the field, where the length of crop lines the vehicle has ahead is not sufficient to extract the perspective model. Further work is directed to solve this problem, and also on the

Fig. 14. Original image the same crop in Fig. 4, but some weeks later.

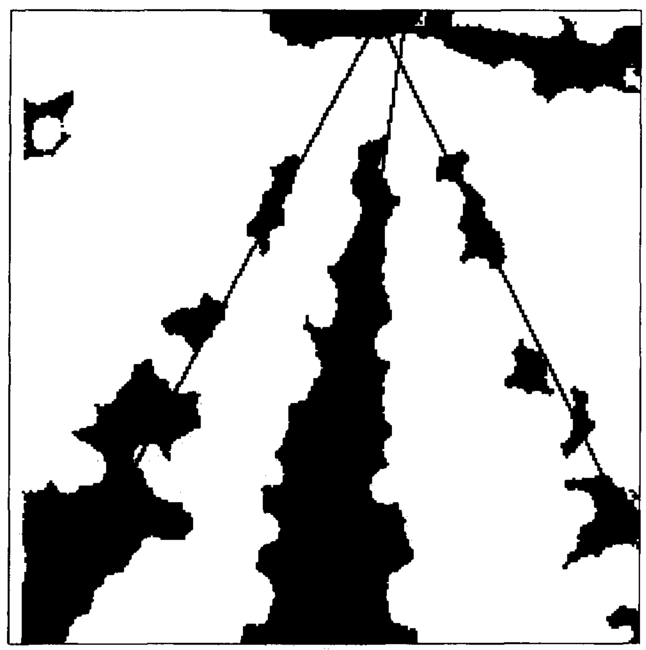

Fig. 16. Perspective model extracted from soil regions in Fig. 15.

implementation of the present method in a parallel machine to achieve real time performance.

Acknowledgments

This work has been partially supported by the BBSRC, and a grant from the Spanish Ministry of Science and Education PF92-52660540.

References

[1] R. Wallace, A. Stentz, C. Thorpe, H. Moravec, W. Whittaker and T. Kanade, First results in robot road-following. Proc. IXth Int. Joint Conf. on Artificial Intelligence, Los Angeles, 2, 1985, 1089–1095.

[2] S.P. Liou, R.C. Jain, Road following using vanishing points. Proc. IEEE Comput. Soc. Conf. on Comput. Vision and Pattern Recogn., 1986, 41–46.

[3] C. Thorpe, M.H. Herbert, T. Kanade and S.A. Shafer, Vision and navigation for the Cangegie-Mellon Navilab. IEEE Trans. Patt. Ana. Mach. Intell., 10(3) (1988) 362–373.

[4] M. Turk, K.D. Morgenthaler, K.D. Gremban and M. Marra, A vision system for autonomous land vehicle navigation. IEEE Trans. Patt. Ana. Mach. Intell., 10(3) (1988) 342–360.

[5] J.D. Crisman and C.E. Thorpe, SCARF: a color vision system that tracks roads and intersections, IEEE trans. Robotics and Automation, 9(1) (1993) 49–58.

[6] R. Schuster, N. Ansari and A. Bani-Hashemi, Steering a robot with vanishing points. IEEE Trans. Robotics and Automation, 9(4) (1993) 491–498.

[7] N. Ayache and O. Faugeras, Building, registering, and fusing noisy visual maps. Proc. Int. Conf. Comput. Vis., 1987, 73–82.

[8] S. Tsuji and J.Y. Zheng, Visual path planning. Proc. 10th Int. Joint Conf. Artif. Intell., 2, 1987, 1127–1130.

[9] D.T. Lawton *et al.* Terrain models for an autonomous land vehicle. Proc. IEEE Conf. Robot. Autom., 1986, 2043–2051.

[10] J.Y. Zheng and S. Tsuji, Panoramic representation for route recognition by a mobile robot. Int. J. Comput. Vision, 9(1) (1992) 55–76.

[11] P.G. Ducksbury, Parallel texture region segmentation using a Pearl Bayes network. Proc. BMVC'93, Guildford, UK, 1993, 185–195.

[12] J.F. Reid and S.W. Searcy, An algorithm for separating guidance information from row crop images. Trans. ASAE, 31 (6) (1988) 1624–1632.

[13] F. Pla, F. Juste, F. Ferri and M. Vicens, Colour segmentation based on a light reflection model to locate citrus fuits for robotic harvesting. Computers and Electronics in Agriculture, (9) (1993) 53–70.

[14] P.C.K. Kwok, A thinning algorithm by contour generation. Comm. ACM, 31(11) (1988) 1314–1324.

[15] F. Pla, Recognition of partial circular shapes from segmented contours. Computer Vision and Image Processing: Image Understanding, 63(2) (1996) 334–343.

[16] M. Straforini, C. Coelho and M. Campani, Extraction of vanishing points from images of indoor and outdoor scenes. Image and Vision Computing, 11(2) (1993) 91–99.

[17] M.J. Magee and J.K. Aggarwal, Determining vanishing points from perspective images. Computer Vision Graphics and Image Processing. (26) (1984) 256–267.

[18] L. Quan and R. Mohr, Determining perspective structures using hierarchical Hough transform. Pattern Recognition Letters (1989) 279–286.

[19] A. Tai, J. Kittler, M. Petrou and T. Windeatt, Vanishing point detection. Image and Vision Computing, 11(4) (1993) 240–245.

[20] R. Dues and A.K. Jain, Clustering methodologies in exploratory data analysis. Adven. Computer, 19 (1980) 113.

[21] R. Brivot and J.A. Marchant, Segmentation of plants and weeds for a precision crop protection robot using infrared images. Proc. IEE Vision, Image and Signal Processing, 1995.

[22] B.K. Jang and R.T. Chin, One-pass parrallel thinning: analysis, properties and quantitative evaluation. IEEE Trans. Pattern Recognition and Machine Intelligence, 14(11) (1992) 1129–1140.