Proceedings of the 1996 IEEE
International Conference on Robotics and Automation
Minneapolis, Minnesota - April 1996

First results in vision-based crop line tracking

Mark Ollis & Anthony Stentz
Robotics Institute
Carnegie Mellon University
Pittsburgh PA 15213

Abstract

Automation of agricultural harvesting equipment in the near term appears both economically viable and technically feasible. This paper describes a vision-based algorithm which guides a harvester by tracking the line between cut and uncut crop. Using this algorithm, a harvester has successfully cut roughly one acre of crop to date, at speeds of up to 4.5 miles an hour, in an actual alfalfa field.

A broad range of methods for detecting the crop cut boundary were considered, including both range-based and vision-based techniques; several of these methods were implemented and evaluated on data from an alfalfa field. The final crop-line detection algorithm is presented, which operates by computing the best-fit step function of a normalized-color measure of each row of an RGB image. Results of the algorithm on some sample crop images are shown, and potential improvements are discussed.

1. Introduction

Agricultural harvesting is an attractive area for automation for several reasons. Human capability is a key limitation in the efficiency of harvesting: for instance, harvesters have been designed which can function at higher speeds than the current standard of 4-5 miles an hour, but the manufacturer found that humans had trouble guiding the machines precisely for long periods of time. In addition, the technical obstacles to automation are less forbidding than in many other areas: the speed of harvesting machines is low, obstacles are uncommon, the environment is structured, and the task itself is extremely repetitive.

The primary perception capability required to automate the harvesting task is detecting the boundary between crop already cut and crop yet to be cut; this boundary is subsequently referred to as a crop cut line. Tracking this line accurately enables the harvester to avoid leaving unharvested crop behind while ensuring that the maximum amount of cutting blade is used on each pass through the field. This paper describes a vision-based crop line tracker which has been used to guide a harvesting machine at speeds of up to 4.5 miles an hour (2.0 m/sec) in an alfalfa field in Hickory, Pennsylvania.

Vision-based guidance of agricultural vehicles is not a new idea, and others have investigated the perception aspect; for instance, Billingsley & Schoenfisch [1] describe an algorithm for guiding a vehicle through row crops. Their algorithm distinguishes individual plants from soil, and has been used to guide an actual vehicle at speeds of 1 m/sec. However, it relies on the crop being planted in neat, straight rows. Reid & Searcy [2] describe a method of segmenting several different crop canopies from soil by intensity threshholding. They do not, however, actually use the algorithm to guide a vehicle. Hayashi & Fujii [3] have used smoothing, edge detection, and a Hough transform to guide a lawn mower along a cut/uncut boundary. Their algorithm only finds straight boundaries, however, and they give no mention of the speed at which they are able to accomplish this task. Jahns [4] presents a review of automatic guidance techniques for agricultural vehicles. To the authors' knowledge, the work described in this paper is the only system which has ever been successfully used to guide a hay harvesting machine.

The testbed used to accomplish this demonstration is Demeter, displayed in Figure 1. Demeter is a New Holland 2550 Speedrower which has been retrofitted with wheel encoders and servos to control the throttle, steering and cutter bar (not shown). A Sun Sparc 20 board running a real time operating system (VxWorks) is used to control machine functions; a separate Sparc 20 is dedicated to the perception system.

Future planned upgrades to this machine include a global positioning satellite (GPS) system and some inertial sensors; these positioning sensors will be used to validate the output of the perception system, and also to help in tasks such as detecting the end of a crop row.

Section 2 briefly summarizes the crop-line detecting techniques which were considered. These methods are primarily of two categories: range-based methods, which attempt to detect the crop height difference between the

0-7803-2988-4/96 $4.00 © 1996 IEEE

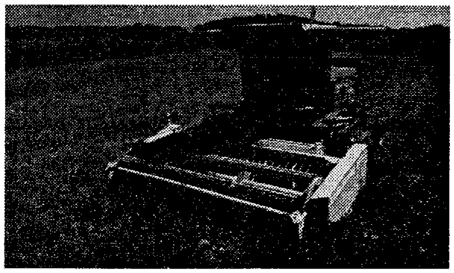

Figure 1: The Demeter testbed.

cut and uncut side, and appearance-based methods, which attempt to detect visual differences between the crop on the cut and uncut side. Range-based methods which were considered include a scanning laser rangefinder, a laser light striper, and a stereo camera system. Vision based methods include a feature tracking technique and segmentation on the basis of texture, intensity, and color.

Section 3 describes in detail the final color-based segmentation algorithm currently implemented on the harvester. The algorithm operates by computing a best-fit step function of a normalized-color measure for each row of an RGB image of a crop cut line. From these step functions, the set of pixels in the image which lies on the crop cut line is then computed. This algorithm is linear in the number of pixels in the row; its derivation is presented in the appendix, and processing results on some sample crop images are shown.

Section 4 shows some results of our algorithm on some images of an alfalfa field in California, and describes some results of field testing the system in Pennsylvania.

Section 5 describes some improvements to the perception system which are currently being investigated. The development of a custom-built camera which provides images at some targeted wavelengths is being investigated as a substitute for the RGB camera, and explicit modeling of the lighting conditions of heavily shadowed fields is being considered to reduce the current algorithms' sensitivity to shadow noise.

2. Candidate crop line detection methods

The class of methods considered for detecting the crop line can be divided into two categories: range-based methods, which attempt to detect the height difference between the cut and uncut crop, and vision-based methods, which attempt to detect appearance differences between the crop on the cut and uncut side. Range-based methods which were considered include a scanning laser rangefinder, a laser light striper, a stereo system, and an optical flow method. Vision based methods include a feature tracking technique and segmentation on the basis of texture, intensity, and color. Preliminary tests of all of the approaches described below were conducted to select a single promising technique to pursue with greater rigor (described in Section 3).

One of the first sensors we considered was a two-axis scanning laser rangefinder. This sensor scans a laser beam over a scene and measures the time it takes for the reflected light to return to the sensor. It returns depth information over a 360 degree azimuth and a 40 degree elevation field of view. The sensor makes about ten sweeps a second; thus, it takes roughly twenty seconds to generate a full 2048 by 200 range image. The scanner is capable of detecting height differences of only a few cm at a range of 20 meters, and in field tests, the height difference between the cut and uncut crop showed up quite clearly. Drawbacks for this system include slow cycle time, high cost, and concerns about mechanical reliability.

We also considered a laser light striping sensor. Light striping systems have two components: a light source, and a camera tuned to detect only the frequencies emitted by the light source. Typically, a laser beam is sent through a cylindrical lens, which spreads the linear beam into a plane. This plane intersects the scene in an illuminated line; distances to all the points on the line can then be estimated by triangulation.

The light striper has no moving parts, requires only minimal computational power, and is potentially inexpensive. However, since we require a system which can function outdoors, a prohibitively powerful laser would be needed to gather range information at the necessary distance (around 5 meters) in order to avoid being washed out by sunlight. The system described above also has the disadvantage that it returns only a linear array of depths, so that only one point along the cut line can be detected at a time.

Stereo cameras provide another possible range-based approach to detecting the crop cut line. Stereo is another triangulation-based method; depth information is obtained by viewing the same object from two or more different viewpoints simultaneously, as with the human vision system. In this application, extremely precise depth information is not needed; we therefore investigated two camera systems with relatively small baselines (between 6 inches and two feet). The Storm stereo algorithm [5] was used to compute the correspondences. Although we found that an alfalfa field contains sufficient texture to solve the correspondence problem, computational speed remains a major concern with stereo-based tracking.

One vision-based technique which showed some initial promise was a window-based feature tracking method [6]. Starting with an image of the crop cut line, a set of small (approximately 20 x 30 pixel) windows which overlapped the cut line boundary were selected by hand. These windows were then input as features to the algorithm, and then tracked from one image to the next. In this manner, it was possible to track the crop line across a sequence of twenty images; however, this method still requires that the initial feature windows be chosen along the crop line.

We also investigated using a local 2-D Fourier transform operator as the basis for texture-based segmentation [7]. The intent was to locate a spatial frequency band with a substantial difference between the cut and uncut crop; however, a preliminary investigation showed no clear evidence of such an indicator.

A visual examination of some crop line images showed that portions of the image containing uncut crop were darker and of a different hue than the portions containing uncut crop. The color difference is generally more pronounced, and is largely due to the exclusive presence of a leaf canopy on the uncut side of the crop. Due to the consistency of this color effect, the relative robustness of the sensor, and the low sensor cost, we chose color segmentation as the method to pursue first for cut line tracking. The implementation of this segmentation is described in the next section.

3. Color segmentation

The color segmentation algorithm has two parts: a discriminant and a segmentor. The discriminant computes a function d(i,j) of individual pixels whose output provides some information about whether that pixel is in the cut region or the uncut region; the segmentor then uses the discriminant to produce a segmentation. Section 3.1 discusses several of the discriminants which were investigated, and Section 3.2 discusses the segmentor algorithm.

3.1. The discriminant

We investigated a number of different discriminant functions d(i,j), such as the percentage intensity within a given spectral band and the ratios between two given spectral bands. In addition to using an RGB camera, we also imaged a crop cut line using a black and white camera with 6 bandpass filters ranging from 300-900 nm: near ultraviolet to near infrared. Each of the filters is transparent to an approximately 100 nm wide frequency band.(The sensitivity of off-the-shelf CCD cameras drops off sharply outside this range).

There are many ways one might attempt to evaluate the relative merit of discriminant functions. In order for the segmentor algorithm to work reliably, it is desirable for the difference in the discriminant between the cut and uncut side to be as large as possible compared to the standard deviation (S.D.) of the discriminant within a given side. Table 1 displays standard deviations and differences between the mean values for several different discriminants from the RGB data, and Table 2 displays the same statistics for some of the best discriminants from the bandpass filter data. The filter data suggest that using a custom-built camera sensitive to only a few narrow frequency bands may provide an advantage over an RGB camera. (These data were calculated from images taken from an alfalfa field in El Centro, CA).

Table 1: RGB discriminants

Discriminant	cut S.D.	uncut S.D.	uncut mean - cut mean
R	22.5	16.8	37.2
G	23.7	20.1	27.9
B	24.3	19.5	31.0
R/(R+G+B)	0.020	0.013	0.033
G/(R+G+B)	0.018	0.007	-0.020
R/G	0.096	0.043	0.137

Table 2: Bandpass filter discriminants

Discriminant	cut S.D.	uncut S.D.	uncut mean - cut mean
650 nm	22.2	8.22	59.1
650 nm/750 nm	0.188	0.040	0.493
550 nm/750 nm	0.112	0.057	0.357

While such statistics are somewhat removed from true performance measures of the system, they do provide some heuristic information. The discriminant used for the images presented here, and also for most of our field tests, was the ratio of red/green; this was selected from the table above and by qualitative observation of a number of segmentations. As we gather a database of images, our plans are to use the Fischer linear discriminant [8] to find the optimal linear discriminant in color space. Using a measure like this on a small set of data may be misleading, however, since we have found that the quality of a given discriminant varies considerably for different images. We have not yet accumulated enough data to determine if any single discriminant will be sufficient for all cases, or whether a system making use of multiple discriminants will be needed.

3.2. The segmentor

For this task, we will confine ourselves to representing segmentations which divide the image into precisely two connected regions. We also assume that the boundary between these regions is a single-valued function of the row coordinate, and that this boundary does not intersect either the left or right edge of the image. This boundary function is represented explicitly by the set of pixels which lie on it, so that nothing further is assumed about the shape of the boundary.

Figure 2 shows a representable segmentation, and Figure 3 shows some non-representable segmentations. This representation was chosen, not because it accurately characterizes all the images which might need to be segmented, but because it can be computed rapidly. Although images such as the ones in Figure 3 do occur, our limited representation covers the majority of the images which appear in a harvesting operation. (Methods for handling other cases are currently under development.)

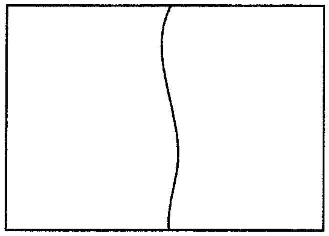

Figure 2: A sample segmentation.

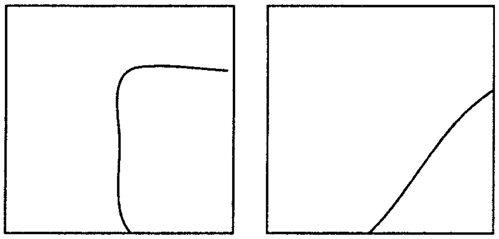

Figure 3: Non-representable segmentations.

The segmentation representation described above lends itself naturally to a scan-line based algorithm, since each image scan line is responsible for contributing one pixel to the segmentation boundary.

Consider a plot of the discriminant d(i,j) of the data along a single scan line (or image row) i. Assume that on the part of the image which contains the cut crop, the d(i,j) will be clustered around some mean value m_c, and the d(i,j) will be clustered around some different mean m_u for the uncut crop. (Empirical evidence shows that this assumption is roughly accurate for most of the discriminants we considered). Under these assumptions, it is natural to try and fit this data with a step function as shown in Figure 4.

Finding the best segmentation is then a matter of finding the best fit step function (lowest least-squared error) to d(i,j) along a given scan line. As we will show, using this definition of best-fit allows us to compute this step function rapidly.

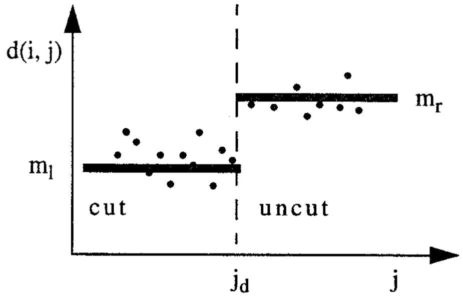

Figure 4: A model plot of d(i, j) as a function of j for a single scan line i.

The step function is defined by three parameters; j_d, the j coordinate of the discontinuity; m_l, the mean value of the step function to the left of j_d; and m_r, mean value of the step function to the right of j_d. Fortunately, it is not necessary to exhaustively search the full three dimensional space to find the best fit; given j_d, it is possible to rapidly compute m_l and m_r. In particular, m_l is merely the average of the d(i,j) on the left side of j_d, and m_r is the average of the d(i,j) on the right.

Thus, an exhaustive search is only necessary across the 1-dimensional space of j_d. The starting point for our algorithm is thus the following:

```
smallest_error = infinity
for each possible $j_d$ from $j_{min}$ to $j_{max}$
  compute $m_l$
  compute $m_r$
  compute error
  if (error < smallest_error)
    smallest_error = error
    best_$j_d$ = $j_d$
  endif
endfor
```

The bulk of the computing time for this algorithm comes from computing m_l, m_r, and the error. Computing these means and errors for the first j_d takes order (j_{max} - j_{min}) time. However, it requires only a small constant number of operations to recompute m_l, m_r, and error for subsequent values of j_d. (See the appendix for the final algorithm and derivation). The end result is that the entire algorithm requires only order (j_{max} - j_{min}) time to find the best fit step function for a given scan line. Using a 400 x 300 pixel window of the image, we currently achieve cycle times of roughly 4 Hz for the entire system.

4. Results

Using the algorithm described above, we have successfully harvested approximately one acre of alfalfa autonomously. This occurred during one week of testing at a site in Hickory, Pennsylvania, in sparse crop, with curved crop lines. (Not all of our testing involved actual harvesting; to avoid cutting all of our available crop, many tests were conducted by simply driving next to the crop line, using the same perception system to track the line.) Our peak speed while harvesting was approximately 4.5 miles an hour; the average speed was approximately 3 miles an hour.

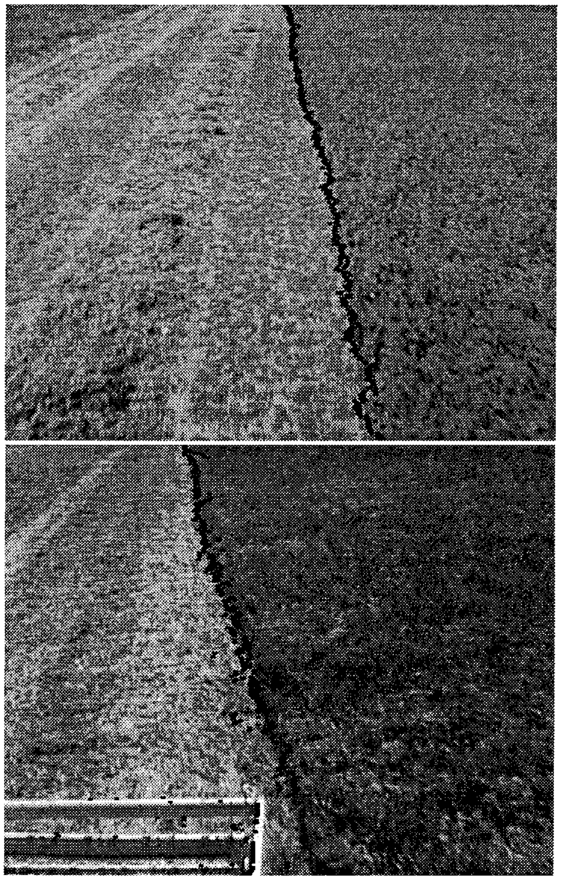

Figure 5: Sample image segmentations

The RGB camera was mounted at the level of the top of the cab (about 4 meters high) and about 2 meters to the side, directly over the crop line. This allowed us to control the harvester without the need for camera calibration; our control algorithm was merely based on steering to keep the cut line in the center of the image. Steering commands were temporally smoothed over a one second time interval to prevent jerky motion. We currently have no quantitative means for evaluating the precision of the cut; however, we estimate that the crop line was tracked successfully to within a tolerance of roughly one foot.

The images in Figure 5 were processed using a discriminant of red/ green. The images shown are 640 x 480 pixels; typically, when running in real time, we use only a 400 x 300 window in the center of the image. The first is from El Centro, California; the second is from our harvester testbed during a tracking run in Hickory, Pennsylvania. The black dots indicate the location of the computed crop cut boundary.

5. Future work

We plan to improve Demeter's perception system in the future on several fronts. First, we are examining improvements to this algorithm such as detecting and removing shadow noise and using a custom-built filtered camera instead of an RGB camera. Second, we plan to integrate the crop line tracker with GPS and inertial sensing in order to provide additional information about the location of the crop cut line and also to help with tasks such as end-of-row detection. Finally, we plan to continue our investigation into alternative ways of sensing the crop line, such as by texture segmentation.

Acknowledgments

The authors would like to thank Nick Collela and Red Whittaker for their technical input; Regis Hoffman and Tom Pilarski for their coding assistance; and Kerien Fitzpatrick, Henning Pangels, and Simon Peffers for assistance with the Demeter harvester experiments. This work was supported by NASA under contract number NAGW-3903.

Appendix: Algorithm derivation

Let j_d be the rightmost column to the left of the discontinuity in the step function, and suppose that the column numbers vary from 0 to j_{max}. Then the m_l, m_r, and error terms can be calculated as functions of j_d as follows (these are defined for j_d from 0 to j_{max} - 1):

$$m_l(j_d) = \frac{\sum_{j=0}^{j_d} d(i,j)}{j_d + 1}$$

$$m_r(j_d) = \frac{\sum_{j=j_d+1}^{j_{max}} d(i,j)}{j_{max} - j_d}$$

$$error = \frac{\sqrt{\sum_{j=0}^{j_d} [d(i,j) - m_l]^2 + \sum_{j=j_d+1}^{j_{max}} [d(i,j) - m_r]^2}}{j_{max} + 1}$$

Clearly, it requires order n time to compute error(j_d) from the d(i,j) alone. It is possible, however, to compute error(j_d + 1) from error(j_d) in constant time. To accomplish this, we express the calculation of error(j_d) in terms of the following sums:

$$t_l(j_d) = \sum_{j=0}^{j_d} d(i,j) \qquad (1)$$

$$t_r(j_d) = \sum_{j=j_d+1}^{j_{max}} d(i,j) \qquad (2)$$

$$t2_l(j_d) = \sum_{j=0}^{j_d} [d(i,j)]^2 \qquad (3)$$

$$t2_r(j_d) = \sum_{j=j_d+1}^{j_{max}} [d(i,j)]^2 \qquad (4)$$

From these, error(j_d) is calculated as follows:

$$error(j_d) = \frac{\sqrt{\left(t2_l(j_d) - \frac{[t_l(j_d)]^2}{j_d + 1} \right) + \left(t2_r(j_d) - \frac{[t_r(j_d)]^2}{j_{max} - j_d} \right)}}{j_{max} + 1} \qquad (5)$$

Since Eqns 1-4 can be computed recursively, it takes only a constant number of operations to compute t_l, t_r, $t2_l$, $t2_r$, and error at j_d+1 given their values at j_d.

We can make the computation faster still by computing, instead of error(j_d), some monotonic 1-to-1 function of error(j_d) such as the following:

$$f(j_d) = [error(j_d) \cdot (j_{max} + 1)]^2 - \sum_{j=0}^{j_{max}} d(i,j) \qquad (6)$$

which can be computed recursively using the following three equations:

$$t_l(0) = d(i,0)$$
$$t_l(j_d) = t_l(j_d - 1) + d(i, j_d) \qquad (7)$$

$$t_r(0) = \sum_{j=1}^{j_{max}} d(i,j) \qquad (8)$$
$$t_r(j_d) = t_r(j_d - 1) - d(i, j_d)$$

$$f(j_d) = \frac{-[t_l(j_d)]^2}{j_d + 1} + \frac{-[t_r(j_d)]^2}{j_{max} - j_d} \qquad (9)$$

This additional speedup results in a roughly 30% increase in cycle rate compared to computing error(j_d) directly from Eqn. 5.

References

[1] Billingsley, J. and Schoenfisch, M. Vision-Guidance of Agricultural Vehicles. Autonomous Robots, 2, pp. 65-76 (1995).

[2] Reid, J.F. and Searcy, S.W. An Algorithm for Separating Guidance Information from Row Crop Images. Transactions of the ASAE. Nov/Dec 1988 v 31 (6) pp. 1624-1632.

[3] Hayashi, M. and Fujii, Y. Automatic Lawn Mower Guidance Using a Vision System. Proceedings of the USA-Japan Symposium on Flexible Automation, New York, NY July 1988.

[4] Jahns, Gerhard. Automatic Guidance in Agriculture: A Review. ASAE paper NCR 83-404, St. Joseph. MI (1983).

[5] Ross, Bill. A Practical Stereo Vision System. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 93), New York, NY June 1993.

[6] Tomasi, Carlo & Kanade, Takeo. Detection and Tracking of Point Features. Technical Report CMU-CS 91-132, Carnegie-Mellon University, Pittsburgh, PA April 1991.

[7] Krumm, John & Shafer, Steven. Segmenting textured 3D surfaces using the space/frequency representation. Spatial Vision, Vol. 8, No. 2, pp. 281-308 (1994).

[8] Young, Tzay & Calvert, Thomas. Classification, Estimation and Pattern Recognition. American Elsevier Publishing Co., 1974, pp. 138-142.