The Demeter System for Automated Harvesting

Tom Pilarski, Mike Happold, Henning Pangels, Mark Ollis, Kerien Fitzpatrick and 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 the Demeter system for automated harvesting. Demeter is a computer-controlled speedrowing machine, equipped with a pair of video cameras and a global positioning sensor for navigation. Demeter is capable of planning harvesting operations for an entire field, and then executing its plan by cutting crop rows, turning to cut successive rows, repositioning itself in the field, and detecting unexpected obstacles. In August of 1997, the Demeter system autonomously harvested 100 acres of alfalfa in a continuous run (excluding stops for refueling). During 1998, the Demeter system has harvested in excess of 120 acres of crop, cutting in both sudan and alfalfa fields.

1. Introduction

Agricultural harvesting is an attractive area for automation for several reasons. Human performance 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 miles an hour, but humans have trouble guiding the machines precisely at these speeds 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.

Demeter, our retrofitted New Holland 2550 Speedrower, has two navigation systems, one camera-based and one based on a global positioning system (GPS). There are several reasons why the use of two separate navigation systems is desirable. Each navigation system has a couple of unique advantages: the camera system can be used without an a priori map of the area and can double as an obstacle detector, while the GPS system is better at preventing positioning error from accumulating indefinitely. In addition, the two systems have quite different failure modes, and so most failures will leave at least one of the systems functioning correctly. For example, GPS is subject to multi-path problems and occluded satellites, while the vision system tends to have trouble with poor lighting conditions and sparse crop. As the navigation systems become more tightly integrated in the future, exploiting the complementary nature of the two will provide a significant increase in overall robustness of the harvesting operation.

The position-based navigation system uses the pose data from the machine controller to guide the machine in the field along planned paths. The pose data is fused together from a differential GPS, wheel encoder (dead-reckoning), and gyro system sensors. The camera system for guiding the harvester consists of three inter-dependent modules: a crop line tracker, an end-of-row detector, and an obstacle detector. The algorithm for tracking the crop line provides the end-of-row detector with information characterizing the difference between cut and uncut crop. The end-of-row detector then acts to constrain the training of crop line tracker. In order for the perception modules to correctly function, an image preprocessor detects and corrects for image distortion caused by shadows.

Figure 1: The Demeter automated harvester.

Neither GPS-based nor camera-based guidance of agricultural vehicles are new ideas. GPS based guidance is a rapidly growing field, particularly in agriculture. For

instance, Larsen et. al.[2] and Stanford University have both demonstrated GPS based guidance of a tractor. Stanford University's system uses four GPS antennas and carrier-phase differential GPS (CDGPS) techniques to compute an extremely accurate estimate of both position and attitude which they use to guide a tractor along a straight line path [1][3]. However, the use of CDGPS requires a method for initializing the system to a correct position solution and hence a pseudolite transmitter is needed. In comparison, the Demeter system only uses one GPS antenna combined with wheel encoder and gyro data to compute estimates of both position and attitude. To the authors' knowledge, the Demeter system is the first to apply GPS based guidance to the task of completely harvesting a field autonomously.

A number of people have investigated camera-based guidance as well. For instance, Billingsley & Schoenfisch [4] 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 [5] 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 [6] 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 [7] and more recently Kondo & Ting [8] present a review of automatic guidance techniques for agricultural vehicles.

2. System Overview

The testbed for Demeter is a New Holland model 2550 Speedrower that has been retrofitted to allow all of the machine actuated functions to be controlled by an on-board computer. Figure 1 shows the actual testbed, while Figure 2 shows a block diagram of the system components. Demeter has two navigation systems, one positioned-based and one camera-based, which it uses to accomplish the task of harvesting a field.

The camera-based navigation system uses sensor information from two color cameras which are mounted on the left and right sides of the Speedrower cab to track a crop cut line, detect the end of a crop row, and detect obstacles in front of the machine. An Intel Pentium-II based PC running the Linux operating system is used as the vision processing computer.

A Motorola MV162 (68040-based) processor board running the VxWorks operating system acts as the physical control computer. It is responsible for combining the differential GPS, gyroscope, and wheel encoder sensor data

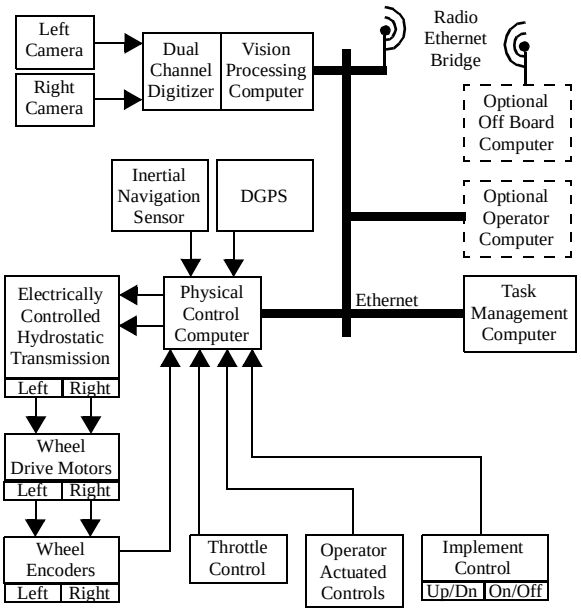

Figure 2: Block diagram of Demeter hardware system

to produce an accurate pose (position and orientation) of the machine at all times. One consumer of the pose data is the position-based navigation system. It combines the pose data with the surveyed geometry of the field to produce and execute motion plans that allow Demeter to completely harvest a field. A second Motorola MV162 processor board running the VxWorks operating system acts as the task management computer on which the position-based navigation system runs.

The modular design of the behavior-based software architecture, shown in Figure 3, provides a flexible system where either the camera-based or position-based navigation system can run separately, or can be combined for a more robust system. The boxes are software modules that run on one of the on-board computers, and circles represent hardware components or sensor data. Communication between modules is represented by lines, where the arrowheads represent the flow of data from one module to another.

The Demeter software system uses a finite state machine to cycle through a series of states, each of which invokes one or more behaviors for execution [9]. The input to the task manager module is a script which defines the states of the finite state machine. Each state contains a list of all software modules that should be running (active) when in this state. The task manager's job is to cycle through the states activating and deactivating the appropriate software module based on "trigger" conditions. Each state contains a list of possible trigger messages that can be received when in this state and the appropriate state to transition to based on the trigger message. An example of a finite state

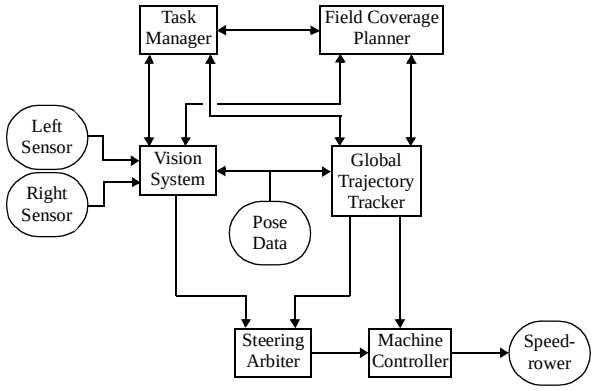

Figure 3: Demeter software architecture

machine which is used to harvest a land section, [the area of uncut crop located between two irrigation borders] is shown in Figure 4. A land section is typically harvested by performing one cut along a border, followed by several center cuts, and then another cut along the last border.

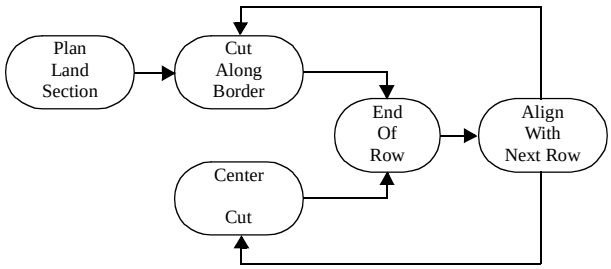

Figure 4: State diagram for harvesting a land section

Constructing a path to harvest the land section is performed by the field coverage planner. The camera system and the global trajectory tracker are used to guide the machine along the planned path. The camera system runs only during the center cuts when there is a crop line to track, and the global trajectory tracker runs during all phases of the path. To resolve any steering conflicts that occur during the center cut, when both the camera system and the global trajectory tracker are guiding the machine, a steering arbiter is used [10]. The steering arbiter receives steering votes from both the camera system and the global trajectory tracker to produce one steering curvature. This steering curvature is sent to the machine controller, which generates analog control signals for the left and right drive motors.

3. Position-Based Navigation

Position-based navigation is accomplished through the interaction between the task manager, field coverage planner, global trajectory tracker, steering arbiter, and machine controller software modules. The two key software modules in the position-based navigation system are the field coverage planner and the global trajectory tracker.

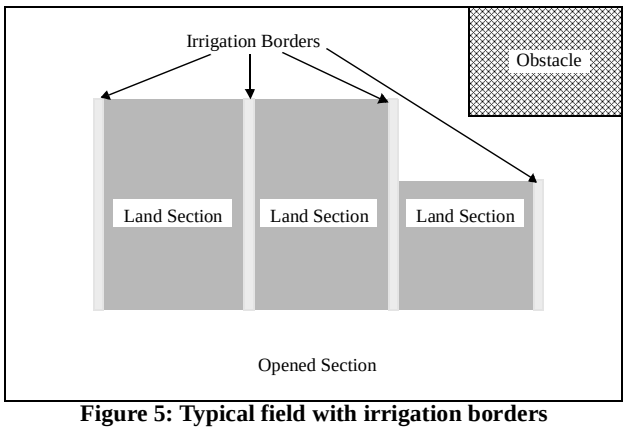

Figure 5: Typical field with irrigation borders

3.1. Field Coverage Planner

Typically, a human operator performs two or three opening passes along the outside borders to "open up" the field (see Figure 5). This gives the human operator room to turn the machine around after coming to an end of a crop row, and to get it aligned with the next crop row. After the field is opened the human operator harvests the field one land section at a time. A land section is the uncut portion of the crop bounded by two irrigation borders. Harvesting a land section requires the human operator to guide the machine down the length of a crop row, detect the end of the row, raise the cutting implement, turn the machine in place, align it with the next crop row, lower the cutting implement, and guide the machine back down the crop row. This process repeats until all land sections are harvested, at which point the field is complete.

Although Demeter is not presently capable of opening up a field, it is capable of harvesting the uncut land sections. This section will define how Demeter constructs and executes coverage plans to accomplish this task.

3.1.1. Surveying a Field

Before Demeter can construct coverage plans for the land sections of a field, it must know where these land sections are. This is accomplished by manually surveying the irrigation borders (four points per border) of the field to obtain differential GPS data points defining the geometry of the field and the locations of the irrigation borders.

3.1.2. Land Section Coverage

In order to harvest a land section completely, a coverage plan must be constructed that moves the cutting implement of the machine over the entire land section. The field coverage planner uses the surveyed geometry of the land sec-

tion along with the geometry of the Speedrower and its implement to construct a coverage plan. The coverage plan minimizes the number of passes required to harvest the land section while trying to keep the cutting width of each pass constant. An example of a coverage plan for a land section is shown in Figure 6.

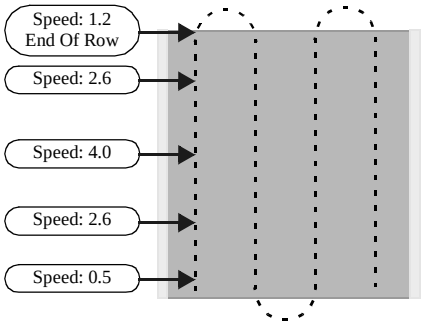

Figure 6: Coverage path for a land section

The coverage plan consists of a coverage path (an array of pose points), a speed profile for the machine along the coverage path, and an array of tagged trigger points along the coverage path. The cutting speed of the machine, which varies depending on the field conditions and the type of crop harvested, is specified in the task manager's script. The speed profile sets desired speeds of the machine along the planned coverage path. Trigger points signal a transition between sections of the coverage path. For instance, each path point that lies on the end of a crop row is tagged as an end-of-row trigger point.

3.1.3. Repositioning Within a Field

Just like a human operator, Demeter must be able to reposition itself within the field. Repositioning is the operation of moving the machine from its current pose to a goal pose while avoiding obstacles and uncut crop within the field. Repositioning is performed by the field coverage planner.

From the surveyed field geometry the field coverage planner constructs a perimeter highway path, shown in Figure 7. The repositioning plans are constrained to use the perimeter highways. Typically, each plan moves the machine along a short path to the main highway, moves it along the highway to an exit point, and then moves the machine along a short path to its goal pose. The plan uses point turns to align the machine with the various path segments and the final pose. An example of a repositioning plan is shown in Figure 8.

3.2. Global Trajectory Tracker

The job of the global trajectory tracker module is to execute all motion plans that the field coverage planner module produces.

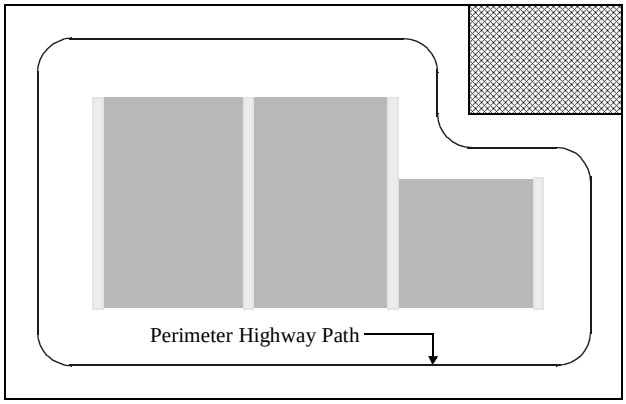

Figure 7: Perimeter highway path

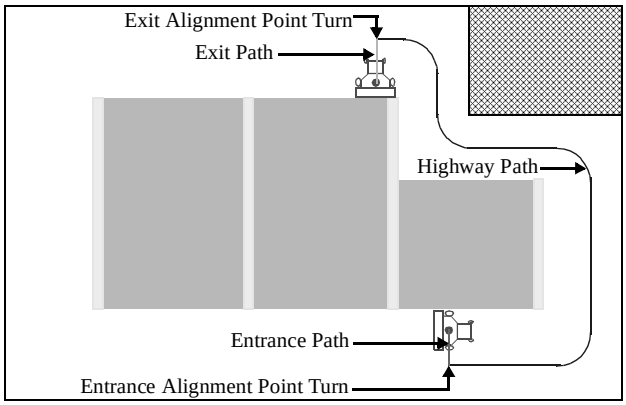

Figure 8: Repositioning example

3.2.1. Executing a coverage plan

During execution of a coverage plan the global trajectory tracker is responsible for generating steering and speed commands that guide the machine along the planned coverage path. Steering commands are generated and sent to the machine controller at a rate of 15Hz using an adaptive pure pursuit algorithm [11]. Pure pursuit calculates the arc the machine must traverse in order to intersect the desired path at point x meters ahead. By varying x, the smoothness of control can be traded off against the precision of the path following. In each control cycle the global trajectory tracker computes the closest point on the planned coverage path to the machine and sends the stored speed command to the machine controller.

In addition to guiding the machine, the global trajectory tracker is also responsible for signaling state transitions along the planned coverage path, by comparing the current machine location to the stored trigger points. For example, for an end-of-row transition, the task manager issues a

command to the machine controller to raise the cutter head of the machine.

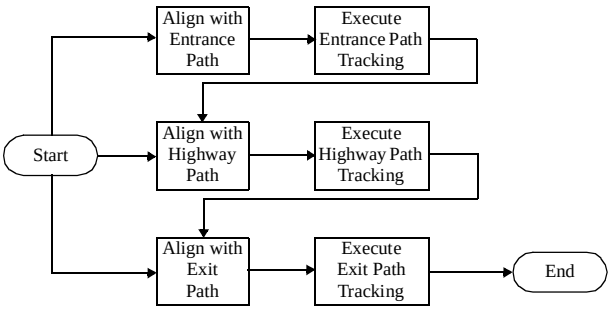

Figure 9: Repositioning plan state machine

3.2.2. Executing a repositioning plan

In addition to all of the responsibilities contained in executing a coverage plan the global trajectory tracker is also responsible for issuing point turn commands to the machine controller when executing a repositioning plan. A point turn has the effect of changing the heading of the machine but not its position. Since the repositioning plan is more complex than simply tracking a single path segment, executing the repositioning plan is implemented as a state machine within the global trajectory tracker, as shown in Figure 9.

3.3. Error Conditions

The position-based navigation system uses the pose data from the machine controller to guide the machine in the field along planned paths. The pose data is fused together from the differential GPS, wheel encoder (dead-reckoning), and gyro system sensors. Although a Kalman filter could have been used for this fusion, the GPS data is typically of high enough quality that the other sensors are merely used to extrapolate the GPS readings (which occur at 5 Hz). Both the dead-reckoning and inertial navigation sensors have a tendency to drift over time resulting in errors in the pose estimate of the machine. Thus, the GPS sensor data is needed to reset the drift in these sensors to produce an accurate pose estimate of the machine at all times. Moreover, if the GPS sensor data is in error, the resulting pose estimate will be in error. Detecting and dealing with erroneous GPS sensor data is performed by the machine controller.

The position accuracy of a GPS system is a function of many variables including the number of satellites in view, the length of the baseline between the base and mobile GPS receivers, and the time delay between differential corrections. The GPS position accuracy of the Demeter system is typically around 10 cm. This is in part due to the relative short baseline (less than 2 miles) between the base and mobile GPS receivers and the multipath-free environment of the test farm.

The GPS receiver maintains its own estimate of the current position accuracy. The machine controller monitors this accuracy and if it is ever found to be above a specified threshold (typically, 20 cm) then a “GPS dropout” is said to have occurred. In response to a dropout, the machine controller stores the current speed and steering curvature of the machine, brings the machine to a halt, and suspends all active software modules. During the dropout, the machine controller continues to monitor the GPS position accuracy until the readings are good again. The controller re-activates the software modules and slowly ramps up the machine’s speed to the desired level.

Since the “pause/resume” implementation of handling GPS dropouts causes the machine to stop motion until the GPS position accuracy is at an acceptable level, it would appear to make the system unproductive, but field testing has shown this not to be the case. Over a thirteen hour period of continuous harvesting, only five GPS dropouts occurred. The length of each pause varied from 30 to 120 seconds and were all caused by a delay in transmission of the differential corrections from the base to the mobile GPS receiver. These delays were primarily due to interference with the radio modems used to transmit the corrections.

4. Overview of the Camera System

The camera system for guiding the harvester consists of three inter-dependent modules: a crop line tracker, an end-of-row detector, and an obstacle detector. The modules share information about the location of the crop to guide their training and segmentation algorithms [12][13][14].

4.1. Crop Line Tracking

The crop line tracking method processes each scan line in the image separately in an attempt to find a boundary that divides the two roughly homogenous regions corresponding to cut and uncut crop. This is accomplished by computing the best fit step function to a plot of a pixel discriminant function f(i, j) for the scan line. The location of the step is then used as the boundary estimate for that scan line.

Figure 10 illustrates this process. Consider a plot of some discriminant function f(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 f(i,j) will be clustered around some mean value m_c, and the f(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 considered). Under

these assumptions, it is natural to fit this data to a step function, as shown in Figure 10. Finding the best segmentation is then a matter of finding the best fit step function to f(i,j) along a given scan line. By defining the best-fit to be the lowest least-squared error, we can compute the step function quite quickly, in time linearly proportional to the number of pixels. Figure 11 shows a sample of a processed image.

After each image is processed, the algorithm updates the discriminant function by computing the Fisher linear discriminant [15] in RGB space between the cut and uncut pixel classes. This function becomes the discriminant used for the next image. The Fisher discriminant computes the line in RGB space such that when the pixel values are projected onto that line, the ratio of average interclass distance to average intraclass scatter is maximized. Intuitively, this results in the linear function which most cleanly separates the cut and uncut pixel classes. Because the initial discriminant function is chosen arbitrarily, a poor choice can result in inaccurate crop line estimates for the first few images until the algorithm converges to better discriminant functions. On average, this results in only a minimal delay. When approaching the end of the crop row, however, the algorithm may diverge from the optimal crop line solution. In Section 4.3., we explain how feedback from the end-of-row detection algorithm is used to solve this problem.

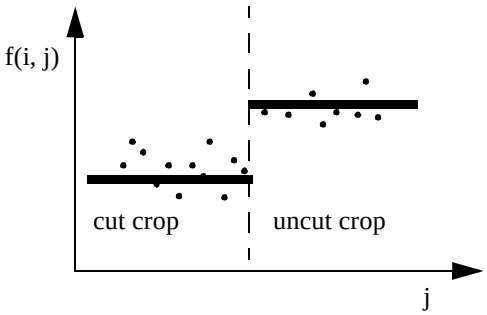

Figure 10: A model plot of f(i, j) as a function of j for a single scan line i.

The presence of more than two homogeneous regions in an image can also cause the crop line tracking algorithm to fail. Wind rows and dried regions of cut crop beyond the first wind row often contribute additional homogeneous regions to the image, and the best-fit step might be found at the boundaries between these wind rows and cut crop. Given prior knowledge of the width of the ideal cut and the perspective projection, these extraneous regions of the image can be masked out to prevent the crop line tracker from locking on to spurious boundaries.

Figure 11: Sample output from the crop line tracker.

4.2. Shadow Compensation

Shadow noise can heavily distort both image intensity (luminance) and color (chrominance). An example of a severe case is shown in Figure 12. Here, a shadow cast by the harvester body lies directly over the region containing the crop line. This shadow is directly responsible for the resultant error in the crop line boundary estimate produced by the crop line tracker.

Shadow noise causes difficulties for a number of reasons. It is often quite structured, and thus is not well modeled by stochastic techniques. Its effects and severity are difficult to predict. If the sun is momentarily obscured by a passing cloud or the orientation of the harvester changes rapidly, the prevalence and effect of shadow noise can vary dramatically on time scales of less than a second.

Normalizing for intensity, though an intuitively appealing method of dealing with shadow noise, fails to be useful in our application for two reasons. The primary problem is that it does not take into account the significant color changes present in shadowed areas. For example, normalizing the image in Figure 12 before processing still results in an incorrect crop line boundary estimate. A number of factors contribute to this color shift, but perhaps the most significant is the difference in illumination sources between the shadowed and unshadowed regions[16]. The dominant illumination source for the unshadowed areas is sunlight, while the dominant illumination source for the shadowed areas is skylight. A secondary problem with intensity normalization is that it prevents the crop line tracking algorithm from using natural intensity differences to discriminate between cut and uncut crop. Depending on

local conditions, such natural intensity differences can be a useful feature.

Figure 12: Shadow noise.

We first present a technique for modeling and removing shadow noise which is based on compensating for the difference in the spectral power distribution (SPD) between the light illuminating the shadowed and unshadowed regions. We then describe a method for adjusting the compensation parameters for individual image characteristics.

In an ideal camera, the RGB pixel values at a given image point are a function of $S(\lambda)$, the spectral power distribution (SPD) emitted by a point in the environment [17]. Our goal is to construct a model of how shadows alter the function $S(\lambda)$.

To a first approximation, $S(\lambda)$ is simply the product of the SPD of the illuminating light, $I(\lambda)$, with the reflectance function of the illuminated surface point, $\rho(\lambda)$. Supposing we assume that every point in the environment is illuminated by one of two SPDs, either $I_{sun}(\lambda)$, comprising both sunlight and skylight, or $I_{shadow}(\lambda)$, comprising skylight only, then the red pixel values for unshadowed and shadowed regions can be computed. In general, however, it is not possible to compute R_{sun} from R_{shadow} without knowledge of the reflectance function of the environment patch being imaged. This is problematic, because for our application, this reflectance function is always unknown. However, if we approximate $\bar{r}(\lambda)$ as a delta function with a non-zero value only at λ_{red}, then R_{sun} and R_{shadow} can be related by a constant factor C_{red}.

The same analysis can be repeated for the G and B pixel values. Under the assumptions given above, the parameters C_{red}, C_{blue}, and C_{green} remain constant across all reflectance functions $\rho(\lambda)$ for a given camera in a given lighting environment. These parameters are used to compensate shadowed pixels.

Determining whether points are shadowed is accomplished by threshholding the red, green, and blue values. Thresholds are set to one standard deviation below the means for each color band. Originally, approximate values for C_{red}, C_{blue}, and C_{green} were hand- selected by experimentation on several images containing significant shadow. For these images, values of $C_{red} = 5.6$, $C_{green} = 4.0$, and $C_{blue} = 2.8$ were found to work well. An attempt was made to calculate C_{red}, C_{blue} and C_{green} values a priori from blackbody spectral distribution models of sunlight and skylight. This calculation produced qualitatively the correct result, e.g. $C_{red} > C_{green} > C_{blue}$. Figure 13 shows an image with shadows removed by means of the above compensation values.

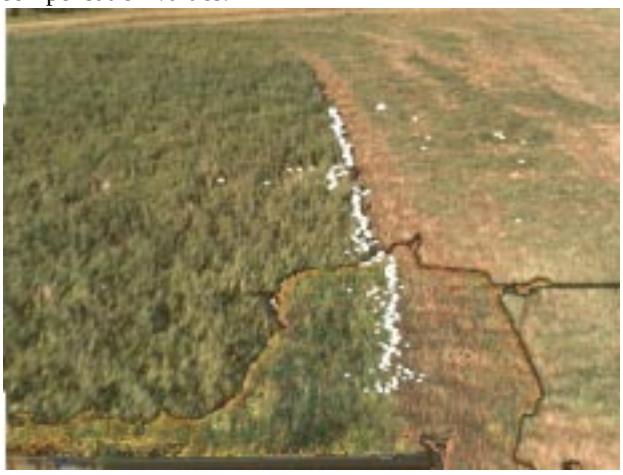

Figure 13: A successful example of shadow compensation.

More extensive experimentation revealed that no single set of compensation values could adequately correct for shadow noise in all images, although the above inequality was found to be generally correct. In fact, a single set of compensation values often could not be found for a given image. Shadows cast by the uncut crop onto itself were the main source of failure, and compensation values tuned to these shadows would subsequently fail to adequately compensate for the shadows falling on cut crop.

To determine whether a single set of compensation values is adequate for a given image, a probability density function (PDF) for the shadowed pixels is built. If the PDF approximates a binomial distribution, the image is considered to require two sets of compensation values. Otherwise, a single set is used.

In addition to the PDF for shadowed pixels, PDFs for unshadowed cut crop and uncut crop are built. These PDFs are used to adjust the proportions of the compensation values, while still maintaining the initial inequality described above.

4.3. Detecting the End of a Crop Row

The goal of the end-of-row detector is to estimate the distance of the harvester from the end of the crop row. When

the end of row boundary is approximately perpendicular to the crop line, and the camera is mounted with zero roll (as in our system), the distance to the end of row is purely a function of the image row where the crop line boundary stops. Figure 14 shows an image which has been correctly processed. The white line marks the computed image row corresponding to the crop row end.

Figure 14: Locating the end of a crop row.

Our end-of-row detection algorithm attempts to find the image row i which most cleanly separates those scan lines containing a crop line boundary from those which do not contain such a boundary.

The results of the end-of-row detection algorithm are fed back to the crop line tracker to constrain its training. Scan lines beyond the end of the crop row are then excluded from the recalculation of the Fisher linear discriminant. This prevents the divergence of the crop line tracker caused by training on pixels that do not reflect the cut/uncut crop separation.

4.4. Obstacle detection

The obstacle detection algorithm is used to locate potential obstacles in the camera's field of view. The method uses a training image to build a PDF for combined cut and uncut crop as a function of RGB pixel value. For each new image, shadows are compensated for as described in Section 4.2. Next, image pixels are marked whose probability of belonging to the crop PDF falls below some threshold. Finally, regions of the image containing a large number of such marked pixels are identified as obstacles. Figure 15 shows an example of such an image before and after processing. Potential obstacles are marked as a solid region.

5. Results

The Demeter system has seen extensive testing in a variety of crops and locations. Crop types include alfalfa (both on

Figure 15: Detecting potential obstacles.

flats and beds) and sudan. The system has been tested on crops in such diverse places as the Imperial Valley, CA, Hickory, PA, and Garden City, KS. Both the camera and GPS positioning systems have made successful cuts running independently, as well as running in conjunction.

In August of 1997, the Demeter system autonomously harvested 100 acres of alfalfa in a continuous run (excluding stops for refueling). This auto-harvest lasted approximately 20 hours, and included operation during night hours. Figure 16 shows the machine in the process of the 100-acre run. During 1998, the Demeter system has harvested in excess of 120 acres of crop, cutting in both sudan and alfalfa fields

When cutting alfalfa, the straight-away speed of the Demeter system equals the average speed of a human operator, which is in the range of 3.5-4.5 mph. Whereas the human operator's performance declines from this peak when cutting in sudan, the Demeter system maintains its high level of performance in this crop. The speed of the system currently does not match human performance in the execution of spin turns. However, the time taken in

Figure 16: The 100-acre cut in progress

spin turns is relatively small compared to the time spent harvesting each row.

The accuracy of the system's cut depends on whether the GPS or camera is being used for guidance, and the performance of each is discussed in the following sections.

5.1. GPS Tracking Results

Of the two methods for guiding a harvester during a cut, GPS has proven to be the more accurate, with accuracy considered as an inverse measure of deviation from an ideal cut. Average error in straight-line cuts is in the range of 4-6 cm. A graph of the deviation of a GPS-guided harvester over the course of a cut is provided in Figure 17.

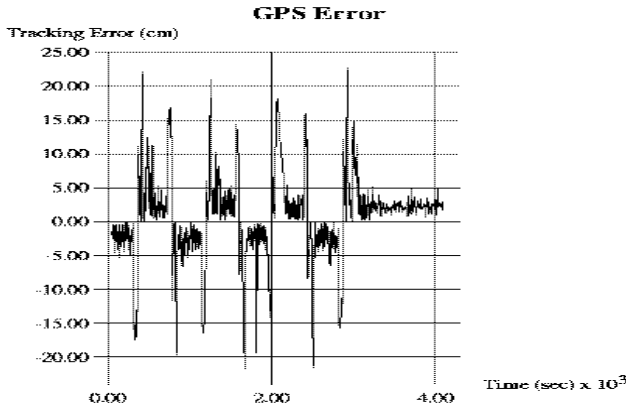

Figure 17: GPS error for a series of cuts

Segments of the graph with low variance correspond to straight-away sections of the cut, with segments of high variance corresponding to executions of spin turns with no cutting. In total, Figure 17 represents a cut of approximately 6 acres.

The accuracy of GPS guidance significantly exceeds the capabilities of even expert human operators, both in individual cuts and in average performance over the course of a working day. Farmers who have assessed the results of the GPS-guided system have uniformly remarked on the perfection of the cut.

GPS guidance has been used to cut over 150 acres of crop, including 80 acres of the 100-acre auto-harvest. Its use has been confined, however, to the flat, rectangular fields found in the Imperial Valley. Future testing will be conducted in hilly terrain and in circular fields.

5.2. Camera Tracking Results

The performance of the camera system varies with crop type, field contours, and lighting conditions. The average error in the cut produced by the camera system falls within a range of 5-30 cm, with the lower end of the range corresponding to ideal conditions: good lighting, a flat field, and thick crop. Because the camera system is following an actual cutline and is bounded by land-section borders, these errors do not tend to accumulate catastrophically, i.e., result in missed crop. Rather, the cut that the system takes becomes progressively larger until the land section is finished; the beginning of a new land section means that the system starts over without the accumulated error of the previous land.

The end-of-row detector has been found to predict the approach of the end of the crop with 90% accuracy. This accuracy must be improved to 100% for the end-of-row to be useful to the system. Further, this accuracy only reflects the lack of false negatives. False positives also occasionally arise, and must be eliminated for the end-of-row detector to be usable.

The camera system has guided a harvester in cutting approximately 80 acres of alfalfa. The camera system accounted for 20 acres of the 100-acre auto-harvest, being shut off for the sake of continued operation during evening hours. Limited cuts in sudan have been made with the camera system, although future work will be required for the system to be able cut a full field of sudan.

6. Conclusions

The success of the Demeter project demonstrates that commercially viable automated harvesting is technically attainable in the near future. As with most automation, robustness to failure and low cost will be the keys to moving the technology into the marketplace. By using two complementary guidance systems made of components available off-the-shelf, we have taken a large step towards the ideal of a truly practical automated harvesting machine.

Future technical work will continue to improve the performance of the individual components. In particular, the obstacle detection capability needs to be substatially improved, since it is crucial for the machine to be com-

mercially viable. The main focus, however, will be on better system integration and graceful failure detection, in order to reduce the impact of inevitable individual component failures.

Acknowledgments

The authors would like to acknowledge Mike Blackwell, Regis Hoffman, Alex Lozupone, Ed Mutchler, Simon Peffers, and Red Whittaker for their work on the Demeter harvester. This work was jointly supported by New Holland and NASA under contract number NAGW-3903.

References

[1] O'Conner, M., Bell, T., Elkaim, G. and Parkingson, B., "Automatic Steering of Farm Vehicles Using GPS," in Proceedings of the 3rd International Conference on Precision Agriculture, June 1996

[2] Larsen, W.E et al. Precision Navigation with GPS. Computers and Electronics in Agriculture, October 1994, vol. 11, no. 1, pp.85-95.

[3] Elkaim, G.H et al. System Identification and Robust Control of Farm Vehicles Using CDGPS. Proceedings of ION GPS-97, Kansas City, MO, September, 1997.

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

[5] 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.

[6] 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.

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

[8] Kondo, Naoshi and Ting, K.C. Robotics for Bioproduction Systems. American Society of Agricultural Engineers, publication 05-98.

[9] Gowdy, J., "SAUSAGES: Between Planning and Action", in Intelligent Unmanned Ground Vehicles, Hebert, M., Thorpe, C., and Stentz, A. (eds), Kluwer Academic Publishers, 1997.

[10] Rosenblatt, J., DAMN: A Distributed Architecture for Mobile Navigation, Journal of Experimental and Theoretical Artificial Intelligence, Vol. 9, No. 2/3, pp. 339-360, April-September, 1997.

[11] Wallace, R., Stentz, A., Thorpe, C., Moravec, H. Whittaker, W., Kanade, T., "First Results in Robot Road-Following," Proceedings of the International Joint Conference on Artificial Intelligence, 1985.

[12] Ollis, Mark. Perception Algorithms for a Harvesting Robot. Ph.D. thesis, Carnegie-Mellon University, 1997.

[13] Ollis, Mark & Stentz, Anthony. First Results in Crop Line Tracking. Proceedings of the 1996 IEEE Conference on Robotics and Automation (ICRA '96), Minneapolis, MN. April 1996, pp. 951-956.

[14] Ollis, Mark & Stentz, Anthony. Perception for an Automated Harvester. Proceedings of the 1997 IEEE Conference on Intelligent Robots and Systems (IROS '97), Grenoble, France. September 1997, pp. 1838-1844.

[15] Duda, Richard and Hart, Peter. Pattern Classification and Scene Analysis. 1973, J. Wiley & Sons, pp. 114-118.

[16] Healey, Glenn. Segmenting Images Using Normalized Color. IEEE Transactions on Systems, Man and Cybernetics, 1992.

[17] Novak, C.L. and Shafer, S.A. Color Vision. Encyclopedia of Artificial Intelligence, 1992, J. Wiley and Sons, pp 192-202.