DEVELOPMENT AND TESTING OF A CRACK SENSING SYSTEM FOR A TETHERED MOBILE ROUTING ROBOT[*]

Lisa S. Matsumoto

DISCLAIMER/DISCLOSURE

"The research reported herein was performed as part of the Advanced Highway Maintenance and Construction Technology Program (AHMCT), within the Department of Mechanical and Aeronautical Engineering at the University of California, Davis and the Division of New Technology and Research of the California Department of Transportation. It is evolutionary and voluntary. It is a cooperative venture of local, state and federal governments and universities."

"The contents of this report reflect the views of the author(s) who is (are) responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the STATE OF CALIFORNIA or the FEDERAL HIGHWAY ADMINISTRATION and the UNIVERSITY OF CALIFORNIA. This report does not constitute a standard, specifications or regulation."

ABSTRACT

As the number of vehicles driving over a particular highway increases, the deterioration of that highway accelerates. Degrading highways require costly maintenance and increase exposure of highway workers to cars traveling at speeds of 100 km/h (65 mph) or higher. For purposes of road crew safety, prolonging the life of the highway, more efficient maintenance operations, and cost reduction, the notion of automating the maintenance tasks was conceived.

Through the joint effort of the California Department of Transportation (Caltrans) and the University of California at Davis, the Advanced Highway Maintenance and Construction Technology (AHMCT) Center has looked into automating the crack sealing process. The Tethered Mobile Routing Robot (TMRR) is designed to locate, rout, and seal cracks, similar to the manual crack sealing operation currently performed by highway maintenance workers.

Crack detection is the most important operation in the overall control scheme of the TMRR. Routing and sealing pavement without the knowledge of the crack's location will make this robot destructive and inefficient. Therefore, the purpose of this thesis is to develop and test a crack detecting algorithm for the TMRR. Based on the results of the tests, the crack finding algorithm demonstrates and validates its practicality for locating a crack in asphalt and concrete pavements. An effective means for tracking cracks in pavement is made possible through the accuracy of the LaserVision and linear positioning systems coupled with the competence of the crack finding algorithm.

EXECUTIVE SUMMARY

Among the many types of highway maintenance tasks, crack sealing is a routine maintenance task performed by a road crew. Cracks are often due to the cyclic loading, incorrect composition of the asphalt mix, shifting or settling of the base, or the contraction or expansion of the pavement due to temperature changes (California Department of Transportation, 1989). Cracks in pavement are a hazard because they allow water and debris to enter the subbase of the roadway. As vehicles travel over these cracks, both water and subbase material are extruded from the cracks due to a pumping action, eventually leading to structural failure.

Through the joint effort of the California Department of Transportation (Caltrans) and the University of California at Davis, the Advanced Highway Maintenance and Construction Technology (AHMCT) Center has looked into automating the crack sealing process. The first all-purpose automated crack sealing machine, developed at the AHMCT Center, is the Automated Crack Sealing Machine (ACSM). The ACSM includes two separate crack sealing machines: one to seal longitudinal cracks, cracks parallel to the lanes, and the other to seal random cracks, cracks generally transverse to the lanes. This report is part of an effort to significantly enhance the ACSM's capabilities through the use of wheeled mobile robot technology. This wheeled mobile robot is designed to locate, rout, and seal cracks, similar to the manual crack sealing operation currently performed by highway maintenance workers, and it is called as Tethered Mobile Routing Robot (TMRR).

However, in the TMRR design, it is possible to consolidate the two distinct machines, the longitudinal crack sealing machine and the general crack sealing machine, into one all-purpose crack sealing robot. The TMRR will replace the two crack sealing machines on the ACSM and will rely on the ACSM to be its support vehicle. The TMRR will have the ability to sense, track, rout, and seal both longitudinal and transverse pavement cracks in one sweep.

Crack detection is the most important procedure in the overall control scheme of the TMRR. Routing and sealing pavement without the knowledge of the crack's location will make this robot ineffective and destructive as opposed to being corrective. Thus, the focus of this research is on the crack sensing and tracking operations of the TMRR. The purpose of this research is to develop the local sensing system on the TMRR and the crack finding procedure.

This document begins with a description of the TMRR followed by a scheme of tasks that are to be performed by the TMRR for successful crack sealing. The document then concentrates on the crack sensing operation. This portion of the document contains a description of the major components needed to carry out the crack tracking operation and the development of the crack finding algorithm. The experimental verification of the crack sensing and tracking operations of the TMRR concludes this section, and the conclusion and recommendations closes the document.

TABLE OF CONTENTS

iv

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

As the number of automobiles in the United States grows every year, the number of vehicles traveling over the same stretch of highway increases accordingly, and each vehicle contributes to the gradual wear of that highway. Therefore, as the number of vehicles driving over a particular highway increases, the deterioration of that highway accelerates. To prevent further degradation and to prolong the life of the road as much as possible, proper highway maintenance is essential. Routine highway maintenance is performed by a road crew usually working in close proximity to cars traveling at speeds of 100 km/h (65 mph) or higher. Highway maintenance workers constantly expose themselves as targets for intoxicated or inattentive drivers behind the wheel of a vehicle traveling at high speeds. According to the August 6, 1989 issue of the Los Angeles Times, more highway maintenance workers in the state of California have been killed on the roads than the members of the California Highway Patrol. As a result, the idea of removing highway maintenance workers from the roads by automating the maintenance tasks was conceived. In addition to eliminating roadway maintenance workers from the highways, automating highway maintenance tasks will shorten the time required to perform highway repairs, alleviating the congestion problems due to lane closures during these maintenance operations. Consequently, Stroup and Velinsky (1994) discovered a significant potential for cost saving due to increased efficiency, increased safety, and fewer traffic delays.

Among the many types of highway maintenance tasks, crack sealing is a routine maintenance task performed by a road crew. Cracks are often due to the cyclic loading, incorrect composition of the asphalt mix, shifting or settling of the base, or the

contraction or expansion of the pavement due to temperature changes (California Department of Transportation, 1989). Cracks in pavement are a hazard because they allow water and debris to enter the subbase of the roadway. As vehicles travel over these cracks, both water and subbase material are extruded from the cracks due to a pumping action, eventually leading to structural failure.

Through the joint effort of the California Department of Transportation (Caltrans) and the University of California at Davis, the Advanced Highway Maintenance and Construction Technology (AHMCT) Center has looked into automating the crack sealing process. The first all-purpose automated crack sealing machine, developed at the AHMCT Center, is the Automated Crack Sealing Machine (ACSM). The ACSM includes two separate crack sealing machines: one to seal longitudinal cracks, cracks parallel to the lanes, and the other to seal random cracks, cracks generally transverse to the lanes. A concise description of the two crack sealing machines is given in the following sections, and the interested reader is referred to Velinsky (1993) and Velinsky (1992) for more detailed information. This thesis is part of an effort to significantly enhance the ACSM's capabilities through the use of wheeled mobile robot technology. To follow, the current crack sealing process will be discussed as it is performed manually, and how the ACSM performs this task.

1.1 Current Caltrans' Methods

The first step in crack sealing is crack preparation. Crack preparation consists of removing loose materials and debris from the crack and possibly forming a reservoir to accept sealant. Removal of loose particles and debris from the cracks is done using a stiff

broom or compressed air. A router is used on narrow cracks to create a larger reservoir to accept the sealant. Routing may produce some spalling along the edges of the crack, however it does not create any problems for the application of the sealant. After the cracks have been cleaned and routed, they are ready for sealing.

The determination to seal a crack is decided by the size of the crack. Caltrans seals cracks that are wider than 6.4 mm (0.25 in), but less than 51 mm (2 in). If the cracks are wider than 51 mm (2 in), then the roadbed is beyond repair, meaning the pavement needs to be rehabilitated (California Department of Transportation, 1989). Extremely narrow cracks are not considered suitable for sealing because routing such narrow cracks would not be beneficial to the pavement. Very narrow cracks are also difficult to track on the pavement surface. On the other hand, extremely wide cracks are usually signs of more serious problems. Wide cracks commonly have depths that extend to the bottom of the pavement and will cause movement of the subbase. With such movement, most available sealants will have a short life. Furthermore, by the time such a crack is found, erosion of the subbase due to the presence of water will have already developed.

1.2 Advancements in Automating the Crack Sealing Operation

As noted previously, the ACSM is comprised of two crack sealing machines to address transverse and longitudinal cracks. The longitudinal crack sealing machine, located on the side of the ACSM, seals cracks running parallel to the roadway while the ACSM moves in the same direction. The machine for transverse cracks operates off the back end of the truck while the vehicle remains stationary. Although the longitudinal and transverse crack sealing machines operate independently of each other, they share the

same support equipment, such as the sealant melter, hydraulic power supply, computers, etc.

1.2.1 Integrated Longitudinal Crack Sealing System

The longitudinal crack sealing system is comprised of the Longitudinal Robotic Positioning System (LRPS), the Local Sensing System (LSS), the Applicator and Peripherals System (APS), and the Integration and Control Unit (ICU). The LSS is a self-contained, compact laser sensor that is used to locate the position and measure the width of the crack. The APS, shown in Figure 1.1, consists of a commercial impact router, burner, blower, debris separator, and the sealant applicator.
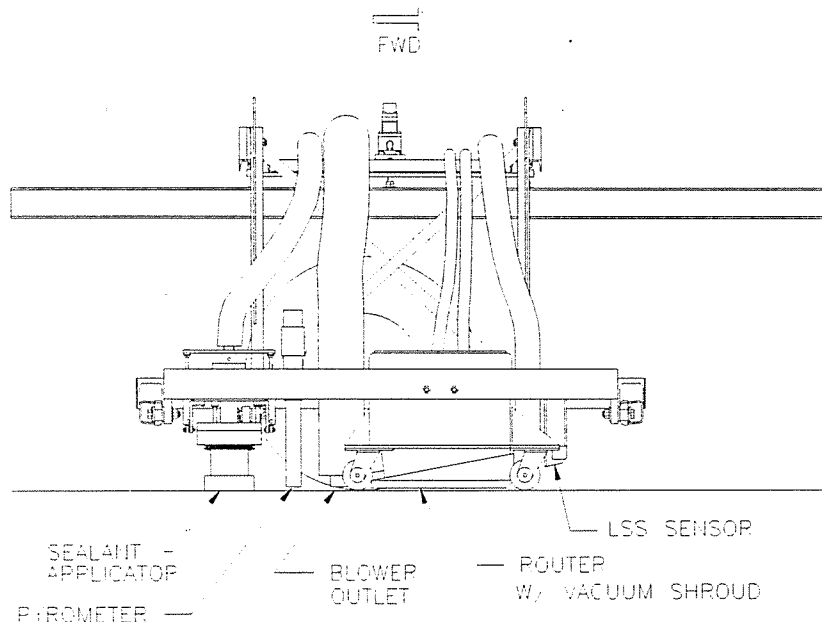
Figure 1.1 *The LRPS with APS and LSS*

The ICU is the main computer which communicates with all of the systems on the ACSM, oversees the entire operation, and coordinates the communication between

systems. The LRPS embodies the LSS and the APS and is supported off the side of the vehicle by way of a simple linkage. The positioning system also provides the motion of the frame normal to the direction of vehicle motion. The strategic placement of this machine allows for easy stowage while not in operation.

During operation, the operator simply needs to maintain the center of the LRPS near the position of the longitudinal crack to be sealed. This can be done visually using a pointing device, or using a camera and monitor. The LSS is responsible for tracking the crack as long as the LRPS is placed over the crack. The LSS obtains the offset of the center of the LRPS to the center of the longitudinal crack. This offset is used by the ICU to modify the desired position input for the closed loop control system. As the crack position is updated continuously, the LRPS is able to track the actual crack position for preparation and sealing.

1.2.2 Integrated General Crack Sealing System

The general crack sealing machine is composed of the Vision Sensing System (VSS), the ICU, the path planning module (which resides on the ICU), the Vehicle Orientation and Control (VOC) system, the general Robot Positioning System (RPS), APS, and the LSS. The VSS is a sensing system based on machine vision used to view an entire lane width and provide crack location and orientation. The VOC system measures the speed and orientation of the vehicle. The RPS positions the APS end-effector according to the path received by the ICU. In this system, the APS is solely the sealant applicator. A passive linkage is used to support the load of the sealant applicator and the LSS. This linkage is driven by the robot arm which is mounted on a linear slide to provide

a redundant degree of freedom. Figure 1.2 displays the integrated general crack sealing system.

As the vehicle moves down the road, the VSS stores a 3.7 x 6.1 m (12 x 20 ft) image of the roadway. The VSS identifies potential crack locations using a path planning algorithm and sends the data to the ICU. The ICU integrates the data from the vision system and VOC system to conduct the path planning process and then transmits the path to the RPS when the identified crack is in the workspace of the RPS. The pre-planned path is used to obtain the starting position of the crack and as a nominal position trajectory. The LSS, which is mounted on the manipulator tooling, is used to verify the presence of a crack and to make the necessary corrections in the crack's planned trajectory.
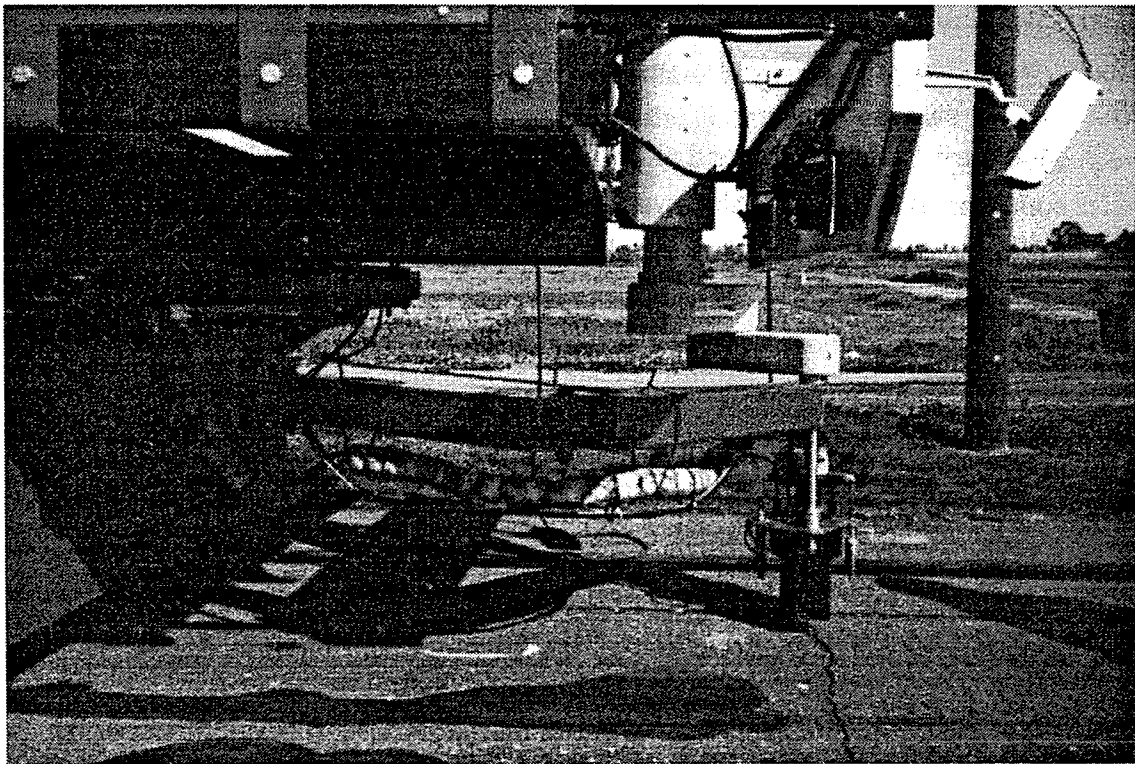
Figure 1.2 *Integrated General Crack Sealing System*

1.2.3 Shortcomings of the ACSM

Because the ACSM was assembled using many commercially available components to improve component reliability and due to the time restriction on the project, there are limitations associated with some of the equipment purchased. For instance, the workspace of the GMF-A510 SCARA manipulator used in the general crack sealing operation is quite limited and does not allow for full lane width operations. Another drawback of commercial robots is their low load carrying capacity. The use of commercial robots for most highway maintenance tasks is not possible because they cannot support large loads such as the weight of many road maintenance devices, such as routers, and the forces that are generated during their operation. Many highway maintenance tasks also require maintaining the device within a specific height off of the pavement, which creates additional complications when using commercial robots.

1.2.4 The TMR Prototype

To overcome the disadvantages of using commercial robots, as noted in the previous section, the concept of a tethered mobile robot (TMR) was developed. The TMR is a self-propelled, robust, controllable wheeled robot working in close proximity to a support vehicle carrying power and supplies. The position of the robot relative to the support vehicle can be measured with high accuracy (Winters and Velinsky, 1992; Velinsky, Hong, and Mueller, 1994). The necessary maintenance supplies and power will be supplied to the TMR through a tether mounted off the rear end of the support vehicle. Figure 1.3 illustrates the overall configuration of the TMR.
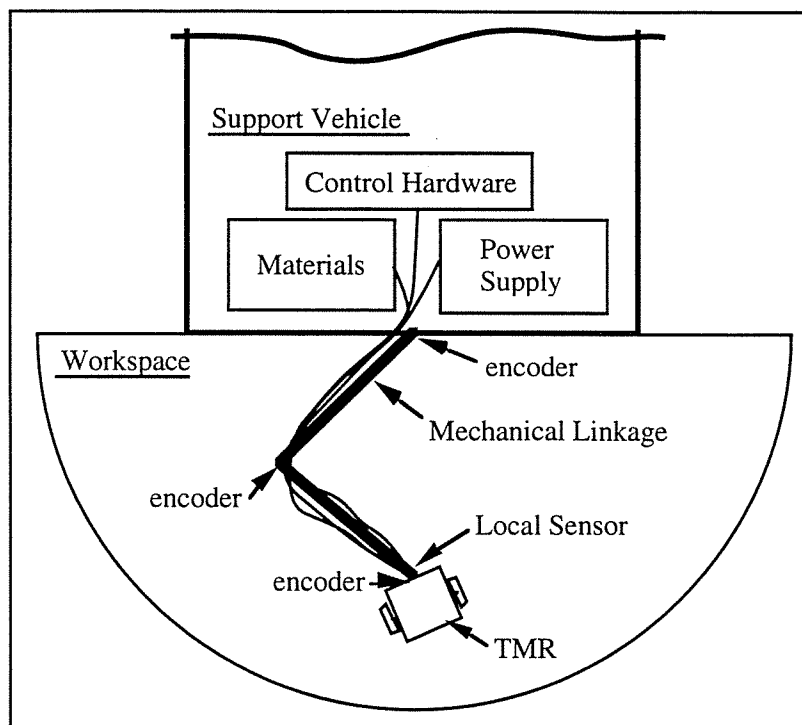
Figure 1.3 *TMR System Configuration*

A full size prototype of the TMR has been designed and built for laboratory use. The prototype measures its position relative to the base plate using one of two methods. The first method uses cable extension transducers (CETs) and triangulation while the second method utilizes the lengths of the passive linkage, which is used to route power and other cables to the TMR, and the angular displacement of each linkage. The current workspace of the laboratory TMR is a semi-circle with a 2 m (7 ft) radius, which exceeds the full lane width of 3.7 m (12 ft). The workspace is defined by the maximum extension of the CETs and the full extension of the linkage. Figure 1.4 displays the layout of the laboratory TMR.
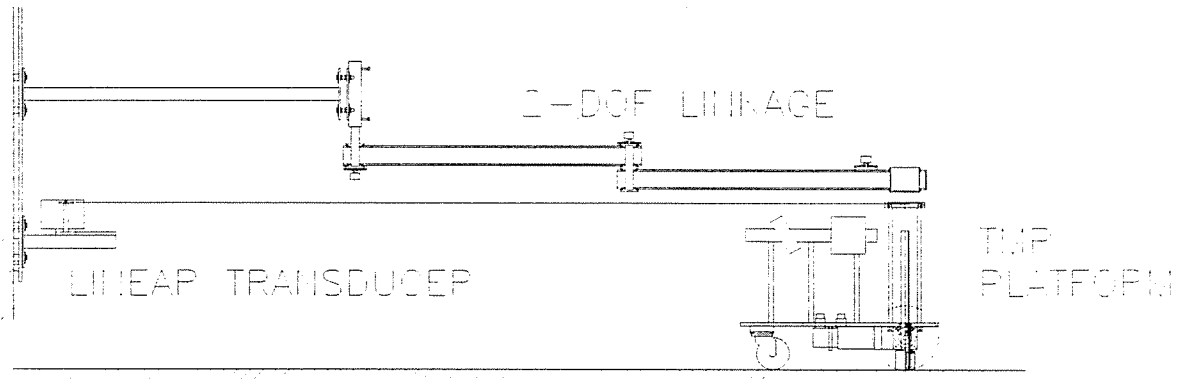
Figure 1.4 *Configuration of the Laboratory TMR*

The prototype operates under several modes: joystick mode, path following mode, and tracking mode. The joystick mode allows the operator to move the TMR using a joystick. In the path following mode, the TMR automatically follows a given path generated by the computer and stored in a file. In the tracking mode, the TMR can track a crack or piece of rope with the use of a laser range finding sensor.

The initial application of the TMR concept was a tool cart that carried the sealant dispenser. Rather than implementing only the crack sealing subtask, the TMR concept has been expanded to implement the entire crack sealing process of locating, routing, and sealing longitudinal or transverse cracks. The notion of performing several subtasks using a single mobile robot is the concept behind the tethered mobile routing robot (TMRR).

1.3 Problem Description and Objective

As mentioned in the previous sections, two distinct machines were necessary on the ACSM to seal the two types of cracks. However, with the development of the TMR concept, the consolidation of the longitudinal crack sealing machine and the general crack sealing machine into one all-purpose crack sealing robot is made possible through the

TMRR. The TMRR will replace the two crack sealing machines on the ACSM and will rely on the ACSM to be its support vehicle like its predecessors. The TMRR will sense, track, rout, and seal both longitudinal and transverse pavement cracks in one sweep.

Crack detection is the most important procedure in the overall control scheme of the TMRR. Routing and sealing pavement without the knowledge of the crack's location will make this robot ineffective and destructive as opposed to being corrective. Thus, the focus of this thesis is on the crack sensing and tracking operations of TMRR. The purpose of this thesis is to develop the local sensing system on the TMRR and the crack finding procedure.

This thesis begins with a description of the TMRR followed by a scheme of tasks that are to be performed by the TMRR for successful crack sealing. The thesis then concentrates on the crack sensing operation. This portion of the thesis contains a description of the major components needed to carry out the crack tracking operation and the development of the crack finding algorithm. The experimental verification of the crack sensing and tracking operations of the TMRR concludes this section, and the conclusion and recommendations closes the thesis.

CHAPTER 2: THE TETHERED MOBILE ROUTING ROBOT

The TMRR evolved through combining the two crack sealing machines on the ACSM and by applying the TMR concept. The TMRR, shown in Figure 2.1, is a general crack sealing machine that will sense, rout, and seal both longitudinal and transverse cracks within a given workspace located behind the support vehicle. This chapter will describe the TMRR and discuss its operation.
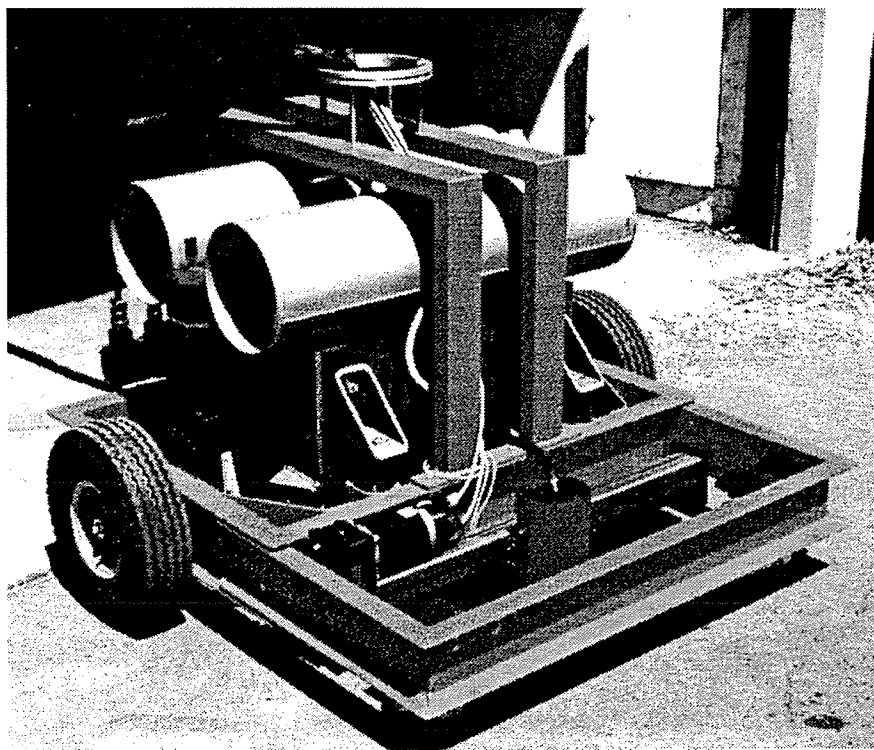
Figure 2.1 *The TMRR*

2.1 General Description

The TMRR will be tethered off the rear of the support vehicle, replacing the general crack sealing machine on the ACSM. The TMRR will be routing the crack in addition to tracking and sealing cracks. All necessary supplies and power will be tethered

to the robot and counterbalanced off the back end of the support vehicle as shown in Figure 2.2.
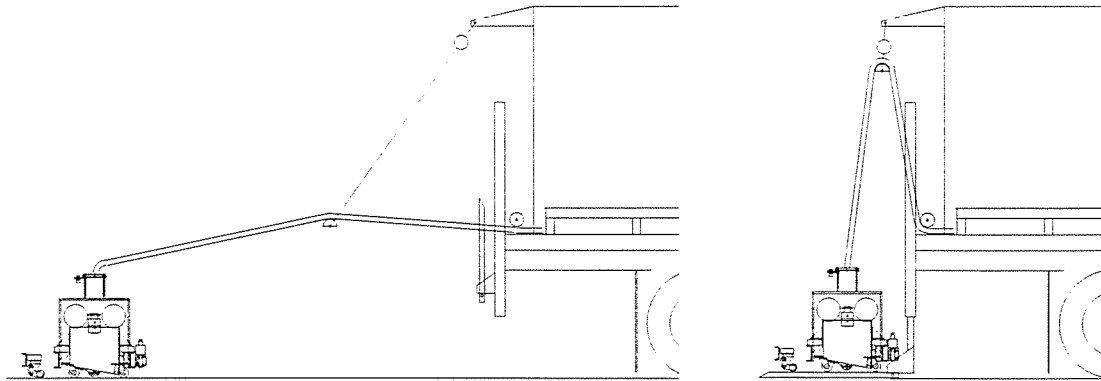
Figure 2.2 *Supplies and Power Routed from the ACSM to the TMRR*

Meanwhile, the position of the robot relative to the support vehicle will be measured accurately through the use of CETs and the application of triangulation.

2.2 Machine Operation

The automated crack sealing process requires some initial human intervention. To initiate the automated crack sealing procedure, the operator must first position the TMRR over the crack using a joystick. The LSS does not have to detect the crack at that time as long as the crack is within the augmented sensing range of the LSS. When the TMRR is placed over the crack, the center of the linear positioner is calibrated, and then the linear positioner sweeps through its full stroke length until the crack is located. Once the presence of the crack is identified, tracking of the crack begins.

2.2.1 Crack Sensing and Tracking

Crack sensing is the most crucial portion in automating the crack sealing process. Any errors generated during the crack sensing process will most likely create problems. For instance, inaccurate position information of the crack will produce two adjoining channels, one created by the router and the other is the crack itself, which weakens the section of pavement located between the two grooves. Such a consequence would require replacing the section of weakened pavement thus causing the maintenance task to being inefficient and more costly.

A crack is detected in the pavement through the integration of the LSS, the linear positioner, and a crack detecting algorithm. Due to the limited 8 cm (3 in) field of view of the LSS, the LSS is placed onto a linear slide with a stroke length of approximately 51 cm (20 in). Further details of the LSS' and the linear positioner's specifications and operation will be given in Chapter 4. Mounting the LSS onto a linear positioner increases the overall crack sensing range to 58 cm (23 in). Figure 2.3 displays the LSS on the linear positioner.

With the aid of the crack sensing algorithm, the position of the crack with respect to the linear positioner is transmitted to the computer as an offset error. The crack detecting algorithm will be discussed in greater detail in Chapter 4.
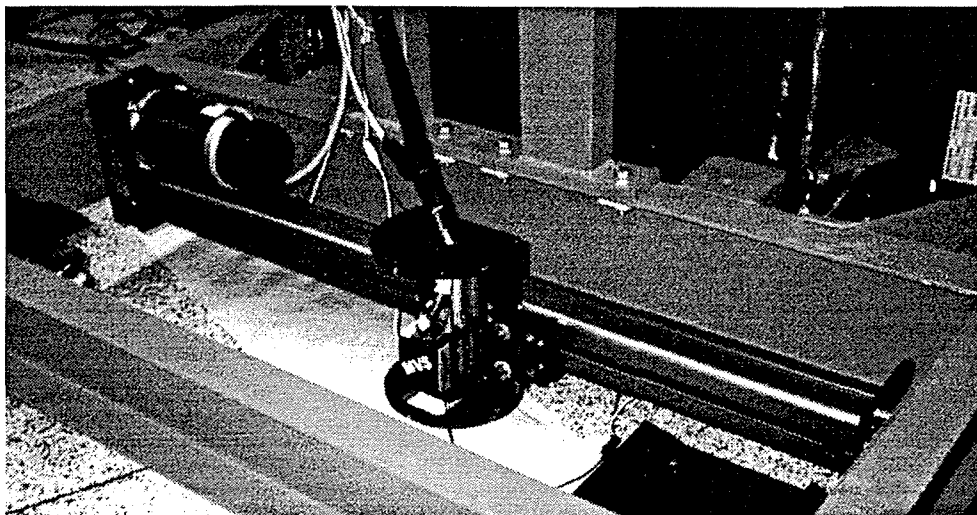
Figure 2.3 *LSS Mounted on the Linear Positioner*

The computer then converts the offset to a "move" command recognizable by the linear positioner's software so that the linear positioner can position the LSS over the center of the crack accordingly. Because of the quick sampling rate necessary for this task, the new crack position is updated constantly in this open loop control circuit. Continuous feedback of the crack position to the computer enables the LSS to track the crack. The location of the crack relative to the center of the TMRR is used to guide the router and to position the sealant head, which is mounted onto the rear of the TMRR, over the anticipated crack position as the TMRR moves over the crack.

2.2.2 Pavement Routing

Once the crack is sensed, its position relative to the center of the TMRR is used to steer the TMRR over the anticipated crack. Routing the crack prior to sealing widens

the sealing reservoir to ensure that a sufficient amount of sealant is to be applied and improves sealant adhesion.

The impact router, that the TMRR will be employing, is the same router that was utilized by the longitudinal crack sealing machine. The router is comprised of a commercial impact cutting wheel that is driven by a hydraulic motor. The cutting wheel holds six rows of cutters evenly spaced about the circumference of the cutting wheel as shown in Figure 2.4 and is designed to run at 2000 RPM.
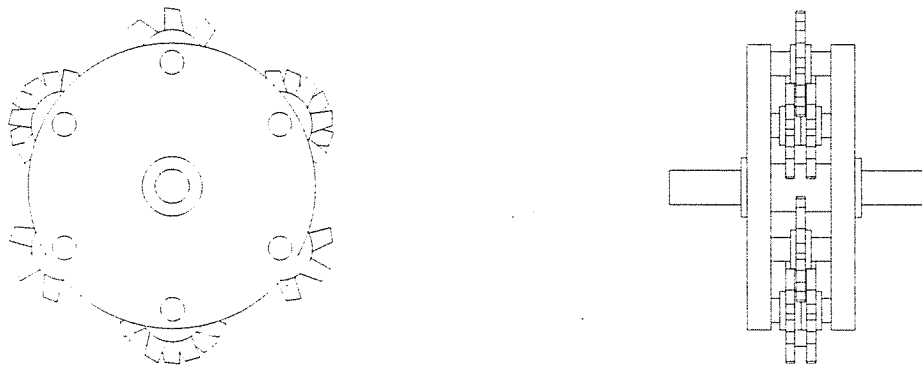
Figure 2.4 *Front and Side Views of the Router Cutting Wheel*

Each cutter has a diameter of 12.1 cm (4.75 in), yielding an overall cutting diameter of 38 cm (15 in). The design of the router's cutting wheel allows for variations in the cutting width and depth which is adjusted by placing the individual cutters in different configurations. The maximum cutting width of this cutting wheel is 5.72 cm (2.25 in), and the maximum cutting depth is 2.2 cm (0.85 in).

The cutting wheel of the impact router will be up-milling while routing pavement. The operation of the router is much smoother while up-milling because the router does

not pull itself out of the pavement when the cutters are operating in this direction. Consequently, greater force is required to drive the router forward since the resulting cutting force acts in the direction opposite to the direction of travel as shown in Figure 2.5.
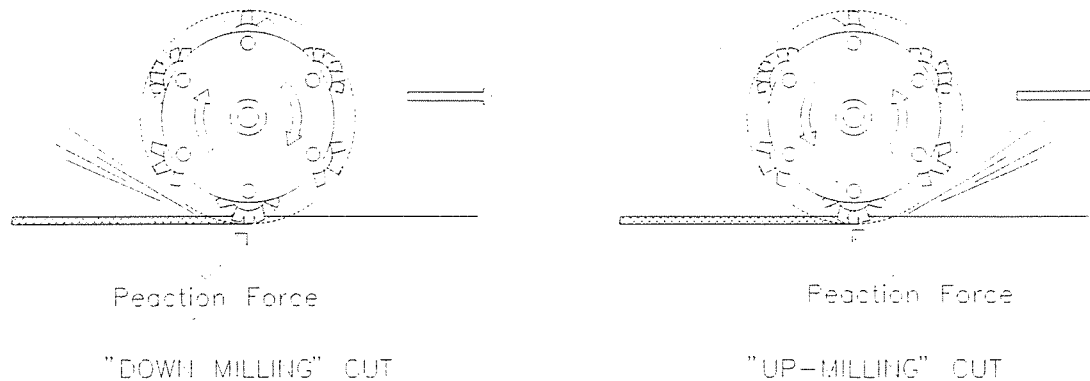
Figure 2.5 *Description of Router Cutting Wheel Operation*

Tests on the router have been conducted to determine the force required to pull the router while up-milling at different speeds. As the cutting width was held constant, the relationship between the force required to pull the router and the router speed at three different depths of cut was investigated. The graphical representation of these results are displayed in Figure 2.6. For this automated highway maintenance task, the speed at which the TMRR will be traveling at should not be any faster than 3 km/h (2 mph). Assuming a cutting depth of 6.4 mm (0.25 in) and moving the TMRR at 3 km/h (2 mph), the force required to move the router is 2520 N (567 lbs).

Router Data Force vs. Speed
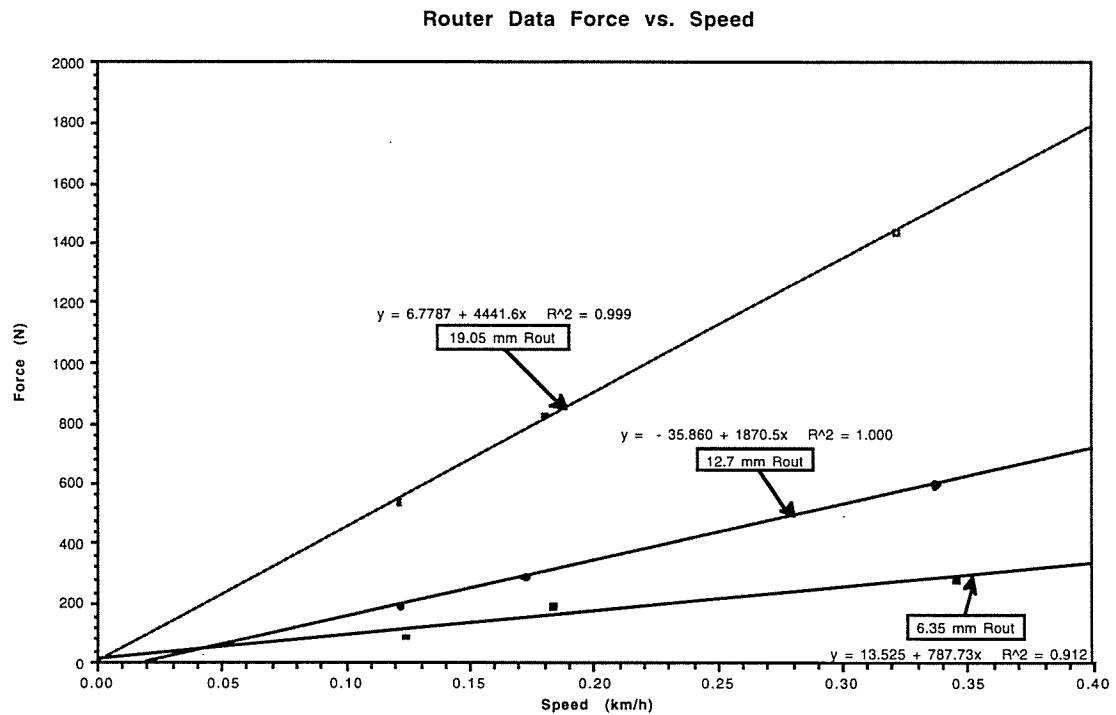
Figure 2.6 *Plot of Router Force Versus Speed*

2.2.3 Crack Sealing

The location of the crack relative to the center of the TMRR is needed to control the angular position of the sealant applicator arm also. The sealant applicator is attached to a 30 cm (1 ft) long arm that extends from the center of the rear of the TMRR at which it is allowed to rotate 180°. A photo of the sealant applicator arm is shown in Figure 2.7.
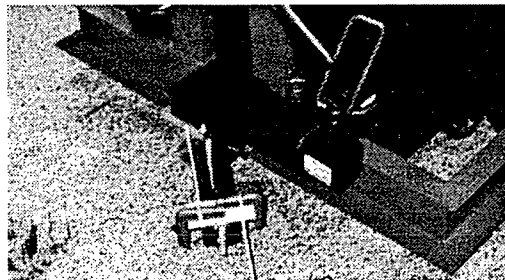
Figure 2.7 *Sealant Applicator off the Back End of the TMRR*

Closed loop position control is used to direct the sealant applicator arm over the anticipated crack as the TMRR continues to move in the direction of the crack. Knowing the orientation of the TMRR, θ, and the position of the center of the TMRR with respect to the workspace represented by (x_o, y_o) in Figure 2.8, the location of the crack with respect to the workspace can be determined.
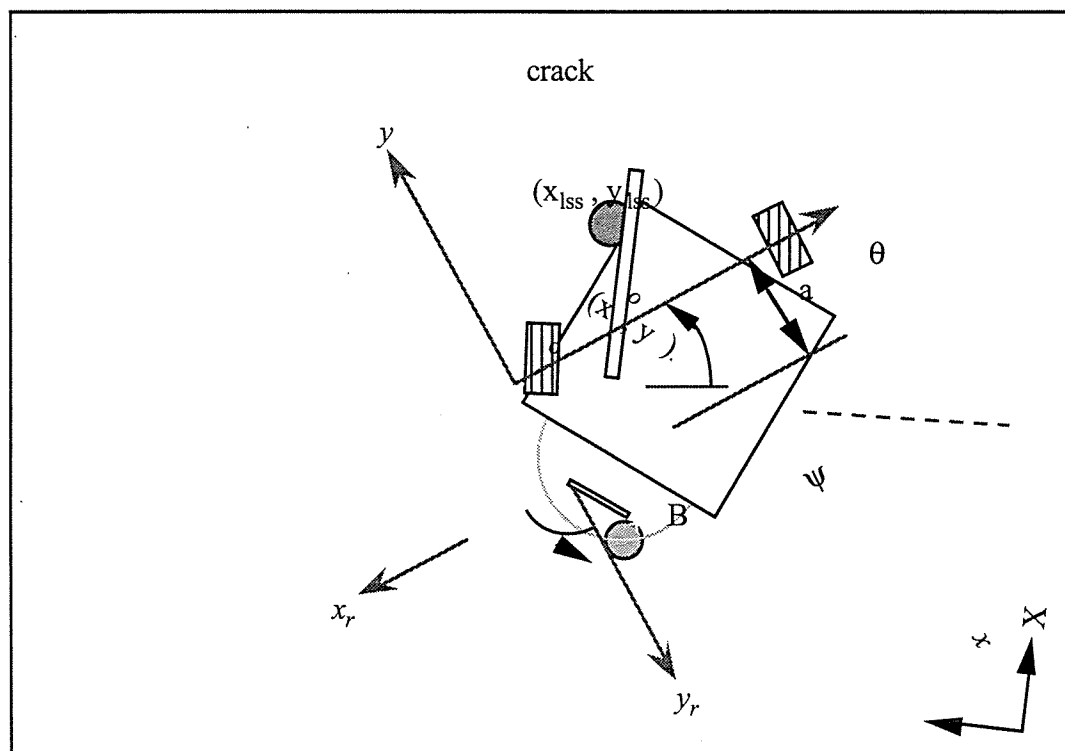
Figure 2.8 *TMRR Over a Crack in its Workspace*

The location of the crack with respect to the fixed frame, (x_{lss}, y_{lss}), is stored in the buffer. To find the intersection point B, a coordinate transformation must be made from the fixed frame of the workspace (X, Y) to the frame fixed to the robot (x, y) using the current orientation, θ. .

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \cdot \qquad (2.1)$$

Then the coordinate frame that is fixed to the robot is translated to the back end of the robot.

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ -a \end{bmatrix} \qquad (2.2)$$

To locate the intersection point B, the radial distance of the crack, r, from the origin of the reference frame fixed to the back end of the TMRR is calculated.

$$r = \sqrt{x_r^2 + y_r^2} \qquad (2.3)$$

The radial distance is then compared to the length of the sealant applicator arm, R. If $R-\varepsilon \leq r \leq R+\varepsilon$, where ε is an allowable offset, then x_r, y_r is an intersection point. When the intersection point is found, the angular position of the applicator arm, ψ, is calculated.

$$\psi = \tan^{-1}\left(\frac{y_r}{x_r}\right) \qquad (2.4)$$

This angular position of the sealant applicator arm is then sent to the sealant arm position controller to properly place the applicator arm over the expected crack.

Through the use of closed-loop proportional control, the applicator provides continuous filling of cracks independent of the crack size and the speed of the applicator. Depending on the width and depth of the crack, various amounts of heated sealant is extruded from the applicator. The flow of sealant is automatically adjusted by a system within the sealant applicator.

2.3 Summary

The TMRR is capable of sensing and tracking, routing, and sealing longitudinal and transverse cracks in pavement. Of the three subtasks that are performed by the TMRR, crack sensing is the most important since proper routing and sealing require accurate crack location information. Knowing the three subtasks that are executed to fulfill the automatic crack sealing procedure, the following chapter will discuss the overall control scheme or the task planning of the TMRR.

CHAPTER 3: TASK PLANNING FOR THE TMRR

The main features of the TMRR are crack tracking, routing, and sealing, with crack detection being the most important. Crack tracking, routing, and sealing are performed automatically in that order once the operator places the TMRR over the crack remotely. The sequence of tasks necessary to carry out the entire crack sealing process is carefully controlled by a main computer program. For the automated crack sealing process to be successful, the location of the crack must be accurately determined because the success of the routing and sealing operations relies heavily on the precise location of the crack.

3.1 Overall Control Scheme of the TMRR

A flowchart of the task planning for the TMRR is displayed in Figure 3.1. The program consists of three parts: (1) the initialization procedure, which includes powering the needed motors and devices, (2) the main body, which executes the three subtasks, and (3) the closing procedure, which returns the TMRR to the base of the truck at the completion of the process.

3.1.1 Initialization

The initialization procedure of the task planning begins with activating the necessary equipment and calibrating the TMRR and the linear positioner.
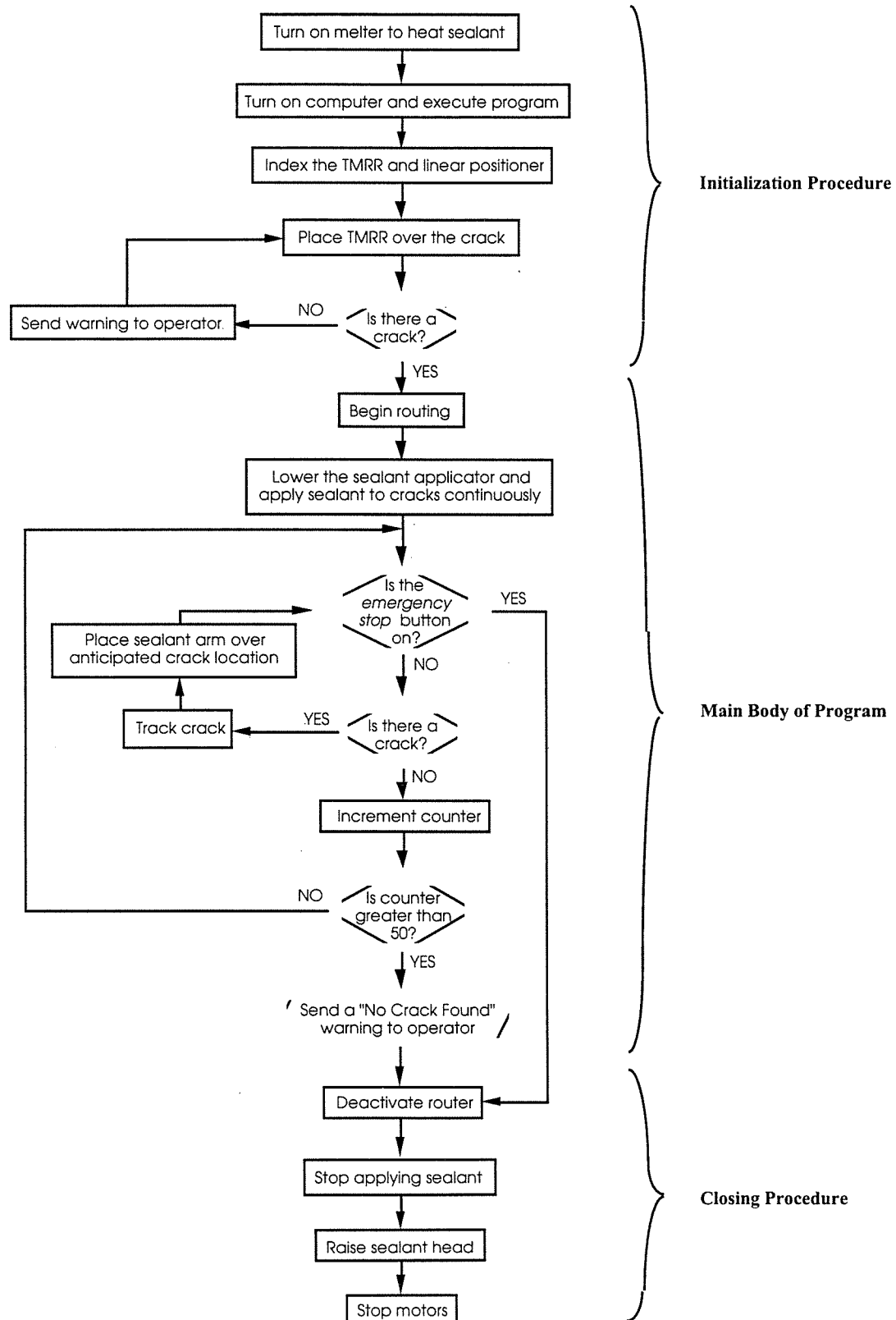
Figure 3.1 *TMRR Task Planning Flow Chart*

The sealant will need to be heated in advanced to ensure that the sealant is at the proper temperature to obtain optimum sealing capabilities. The computer is powered, and the main program is executed to run the initialization procedure. Prior to activating the rest of the equipment, the hydraulic lift lowers the TMRR to the ground as shown in Figure 3.2, and the operator pulls the TMRR off of the lift. The operator then sets the TMRR to its index or "home" position by fitting the TMRR into the base plate, which is mounted beneath the bed of the truck.
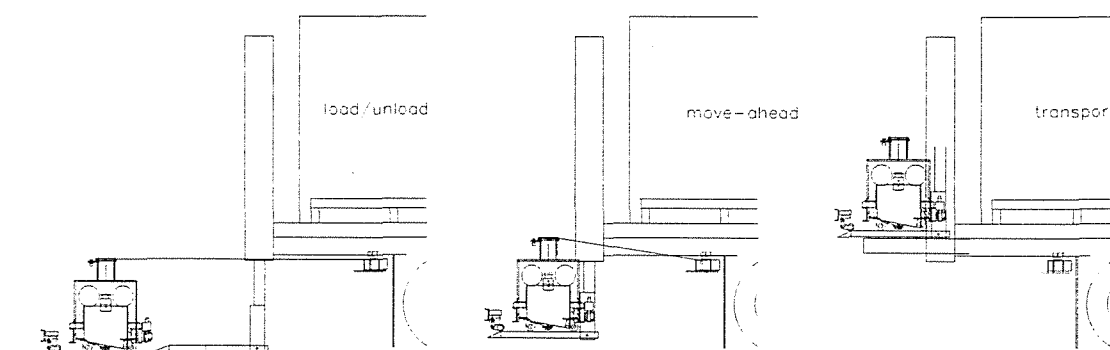
Figure 3.2 *TMRR on a Hydraulic Lift Mounted off the Back End of the ACSM*

Once the TMRR is calibrated, the rest of the equipment is powered. Although the hydraulic motor on the router is powered, routing of the pavement is prevented by retracting the cutting wheel. The program will be in joystick mode to allow the operator to move the TMRR to the start of the crack in an orientation generally along the crack. The crack does not have to be detected by the laser sensor at the time the TMRR is placed over the crack. As long as the crack is within the augmented field of view of the laser sensor, the crack detecting routine should be able to identify the location of the

crack. The linear positioner is then calibrated to its "home" position, which is aligned with the center of the TMRR. The linear positioner then moves the laser sensor to one end and begins searching for a crack using the crack detecting routine, which will be described in greater detail in the following chapter. When the crack is detected, initialization is complete.

3.1.2 Main Body of the Program

Once the initialization portion of the task planning is done, the TMRR is ready to track, rout, and seal the crack automatically. During this portion of the program, if the operator wants to stop the crack sealing process at anytime, he can hit the emergency stop button to deactivate the application of sealant, the router, and the motors on the TMRR. With the laser sensor positioned over the center of the crack, tracking of the crack begins. The position of the crack relative to the center of the TMRR is transmitted back to the computer to steer the impact router over the crack. The TMRR continues to track the crack until the detected crack is under the router. The TMRR then remains stationary while the cutting wheel of the impact router is extended towards the ground to obtain the desired cutting depth. When the desired routing depth is obtained, the sealant head is lowered to apply sealant as the detected crack appears in its workspace. The position information is also sent to the sealant applicator such that the sealant head can dispense sealant into the routed crack as the routed crack enters the workspace of the sealant applicator. The TMRR continues to track, rout, and seal the cracks until the end of the crack is reached or the boundary of the workspace is detected.

3.1.3 End of Program

The crack repairing operation ends when the end of a crack is located or the TMRR is on the verge of extending past the limits of its workspace. When either of these events occur, the router will be retracted first and then shut off as the TMRR slows down and continues to move forward. Then, the sealant dispenser will be turned off to ensure that enough sealant was administered to seal to the end of the rout. All the while, the TMRR decreases its speed gradually while the sealant head is still in the lowered position to scrape off any excess sealant. After a short amount of time, the sealant head is then lifted off the ground and then the TMRR returns to the rear of the truck where it places itself on the hydraulic lift automatically to be transported.

3.2 Summary

The task planning or control scheme for the TMRR consists of a sequence of tasks to be executed in order to repair cracks successfully. Out of all of the tasks that are executed, crack sensing is the most important because the success of the other operations depends on the accurate position of the crack with respect to the center of the TMRR. The next chapter describes the previous methods used to identify cracks in pavement and the current crack sensing method utilized by the TMRR.

CHAPTER 4: THE CRACK DETECTION SYSTEM FOR THE TMRR

Crack sensing is the most crucial operation of the task planning for the TMRR. The success of the routing and sealing operations is governed by the accuracy of the location of the crack with respect to the center of the TMRR and the rate at which the crack's location can be determined. This chapter will examine the previous approaches to automated pavement crack detection and the crack sensing approach to be implemented by the TMRR.

4.1 System Components

There are two systems that compose the crack detection system for the TMRR: the local sensing system and the linear positioning system. The local sensing system relies on the linear positioning system to augment the sensor's field of view and to continuously place the sensor over the center of the crack.

4.1.1 The Local Sensing System

Detecting the presence and position of the crack is made possible with the LSS once utilized by both the longitudinal crack sealing machine and the general crack sealing machine. The LSS is a LaserVision sensor, model MVS-30, manufactured by Modular Vision Systems Inc. (MVS). The LaserVision sensor is displayed in Figure 4.1. This laser sensor was specifically designed for tracking and inspection in robotics applications. The sensor is compact, easy to use, and robust in harsh environments.
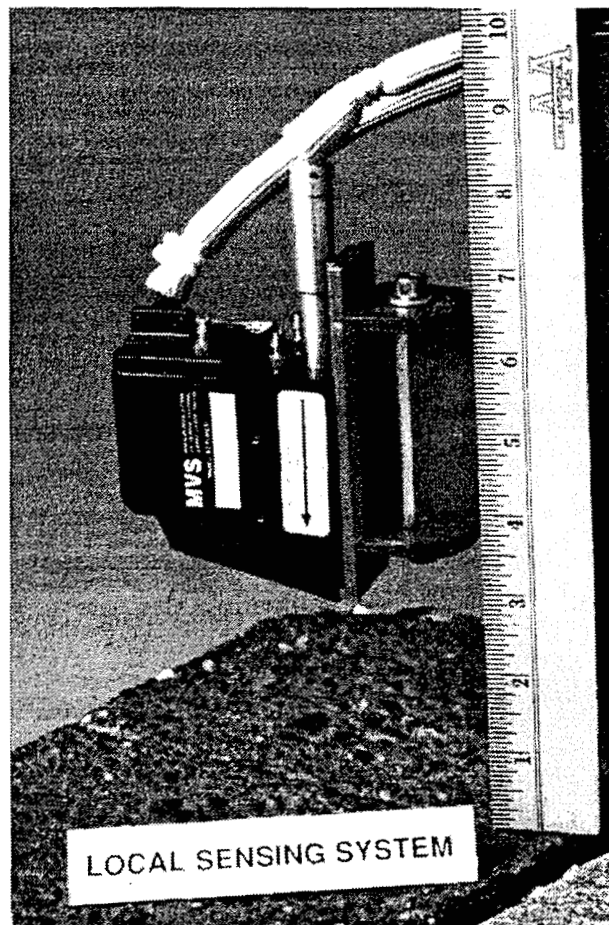
Figure 4.1 *The LaserVision Sensor*

Laser sensors are not sensitive to color variations and dust, which make them a suitable candidate for our application.

4.1.1.1 Component Description

The MVS LaserVision System consists of the laser sensor, the related hardware and software, and a power supply. The LaserVision sensor is contained in a compact 10 x 7.6 x 4.1 cm (4.0 x 3.0 x 1.6 in) casing, weighing 0.255 kg (0.562 lbs). Within this

package is a laser light source, a CCD camera, and appropriate optics. Hardware and software are included in the MVS LaserVision system and are listed in Table 4.1.

One LaserVision Sensor
One 30 mW laser source
One sensor and laser power supply
One LaserVision Image Processor
One coprocessor board
One Microsoft compatible C5.1 library of driver routines

Menu Driven Program
     - Profile capture, store, segment, recall
     - Adjust laser intensity

Segmentation Program

IBM compatible 386-25 MHz
     - VGA color monitor and graphics card
     - Four empty full sized slots
     - Two serial ports
     - 40 MB hard drive
     - 4 MB RAM

Table 4.1 *MVS LaserVision System's Software and Hardware*

The power supply is a standard rack mounted power supply which provides power for the laser and the camera. Refer to Appendix A for a more details on the LaserVision sensor.

4.1.1.2 Specification of the LaserVision Sensor

The sensor requirements for the previous crack sealing operations coincide with our specifications, and these requirements are summarized in Table 4.2.

RESOLUTION ALONG SCAN	1.59 mm (0.0625 in)
VERTICAL RESOLUTION	1.59 mm (0.0625 in)
ACCURACY OF CRACK POSITION	3.1750 mm (0.125 in)
FIELD OF VIEW	10 cm (4 in)
DISTANCE TO SURFACE	10 cm (4 in) minimum
SYSTEM RESPONSE FREQUENCY	18 Hz
HUMIDITY	0 to 85%
VIBRATION	3 g peak from 15 Hz to 100 Hz
SHOCK	10 g
OPERATING TEMPERATURE	0.0 to 71.1°C
SERVICE LIFE	10 years
SENSOR MUST ENDURE	• wind and sunlight • dusty environment • surface color variations • moisture on pavement • debris in cracks • road surface height variations • temperature variations • electromagnetic interference
SENSOR MUST DISTINGUISH BETWEEN	• previously filled cracks • oil spots • shadows • actual cracks

Table 4.2 *Sensor Requirements Obtained from Krulewich and Velinsky (1992)*

The requirements for the sensing system are based upon a response time fast enough to track cracks at speeds up to 3 km/hr (2 mph). A sampling rate of 18 Hz is necessary in order for the sensor to not lose view of the crack while traveling at 3 km/hr

(2 mph). The sampling rate includes the time required to acquire the profile of the crack and the processing time associated with identifying the crack. We have also assumed that the typical random crack do not vary by more than 45°.

To the accommodate for possible violent vibrations generated by the router, the laser sensor has been lowered slightly, decreasing the field of view to 8 cm (3 in) instead of 10 cm (4 in).

Appendix A contains the manufacturer's specifications for the MVS-30.

4.1.1.3 Operation of the MVS LaserVision System

The LaserVision sensor uses the principle of triangulation to measure range information. Distance measurements are made by projecting a light source onto a diffusive surface and the deflected light from the surface is imaged onto a linear position detector or linear diode array where the displacement of the image can be read. See Appendix B for a more detailed description of optical triangulation. The LaserVision sensor projects a line of laser light onto the pavement surface and the CCD camera is used to observe the line generated by the sensor. The CCD camera output is the input to the MVS LaserVision Profile Board LPB-200, an image processing board. A three dimensional profile of the pavement surface is then extracted through the use of the LPB-200.

4.1.1.4 Data Obtained by the LaserVision Sensor

The LaserVision sensor is set up in such a way that a triangular plane of laser light is emitted from the laser head. At the intersection of the planar laser light and the plane of the ground, a line of laser light appears, as illustrated in Figure 4.2.
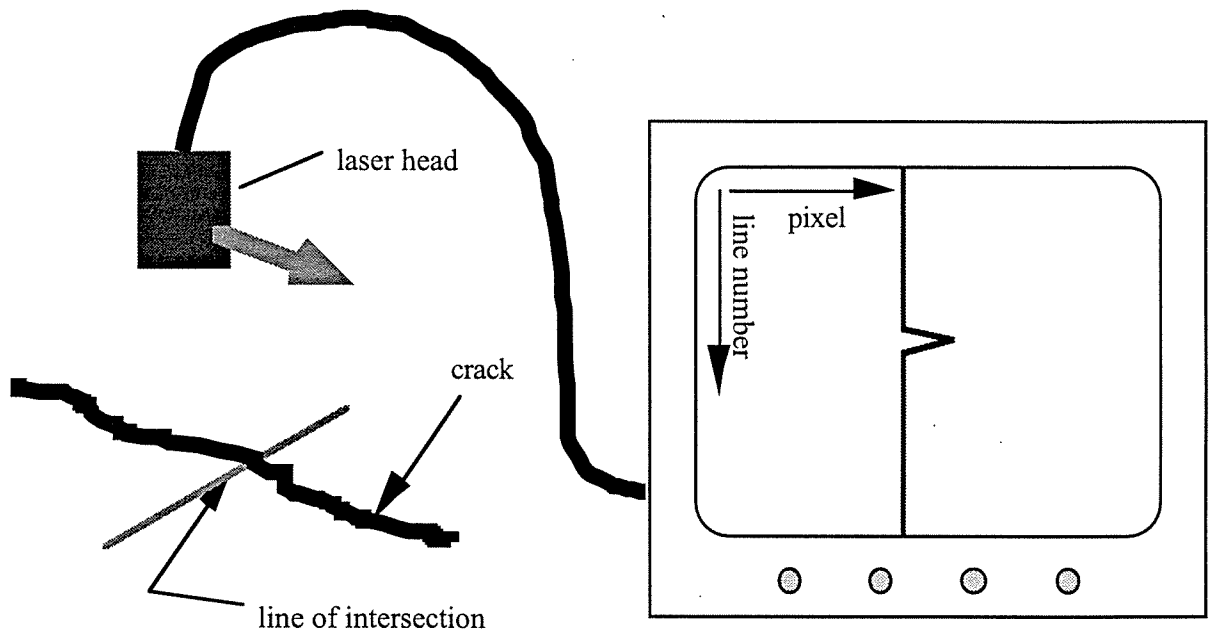
Figure 4.2 *Laser Sensor Acquiring a Pavement Crack and Displaying it on the Monitor*

The camera, which is a part of this laser sensing system, captures the profile of the projection of the laser beam onto the ground and displays it onto a black and white monitor, as shown in Figure 4.2. Using the software, that was supplied by MVS, the pixel, line number, the position across the profile, u, and the depth at that point, v, can be extracted. The value of the pixel is given along the x-axis and the line number is given along the y-axis of the monitor with the origin located at the upper left hand corner. This is also shown in Figure 4.2. The camera coordinates are converted into user coordinates u and v whose units are in millimeters. The orientation of u and v are shown in Figure 4.3.

The zero coordinate along the *u* axis is the center of the laser scan, while the zero

coordinate along the *v* axis is the surface of the pavement.
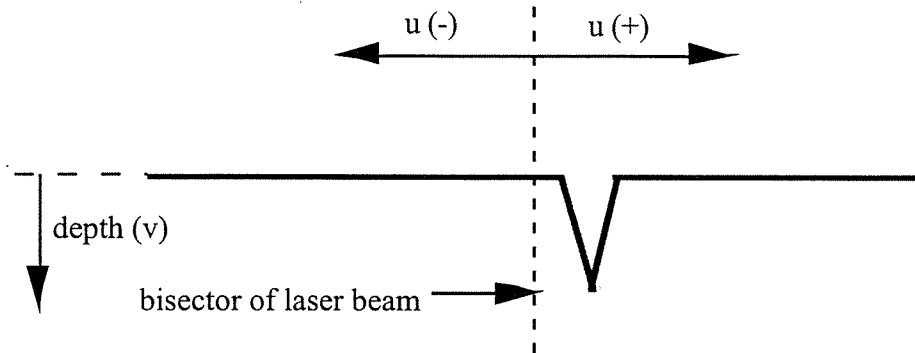
Figure 4.3 *Coordinates of Acquired Crack Profile*

4.1.2 The Linear Positioning System

Because of the limited field of view of the laser sensor, the decision was made to

mount the laser sensor onto a linear slide or a linear positioner to increase its field of view.

To avoid compatibility problems, rather than purchasing the linear positioner, encoder,

motor, and motor controller separately, a vendor selling an integrated system was sought.

4.1.2.1 Specification of the Linear Positioning System

Due to the demand for accurate crack position information, the linear positioner

should be repeatable with a high positioning resolution. A high resolution encoder and/or

a step motor is commonly used for position feedback.

The conditions and environment influenced the requirements for the linear

positioning system. Due to the availability of power on the support vehicle, a linear

positioning system that was powered electrically was preferred over a pneumatically

driven system. Because of the routing operation, the linear positioning system will be operating in a dusty environment. Therefore sealing the system from dust is desirable.

Constrained by the size of the TMRR's frame, the length of the linear postioner with the motor attached should not exceed 0.9 m (3 ft). Therefore, to maximize the stroke length, the motor needs to be mounted above or below and parallel to the linear positioner. A stroke length of 61 cm (24 in) was selected. Limit switches would be required to mark the ends of the stroke. The linear positioner should also be able to support the weight of the laser sensor and its cover.

Assuming the TMRR is traveling at its maximum speed of 0.9 m/s (2 mph), the maximum stoke speed was calculated to be 0.46 m/s (18 ips). The calculations performed to obtain the maximum stroke speed can be found in Appendix C.

Jasta-Dynact was the only manufacturer that could provide us with an integrated linear positioning system that met our specifications, and their LP-150 model is used for our application. Additional information and manufacturer's specifications for the components that comprise the system are given in Appendix D.

4.1.2.2 Operation of the Linear Positioner

The motor control uses Intelli-Command Language (ICL) as its operating language. The ICL system includes commands, parameters, and user variables that allow the user to perform motion. A sequence of ICL commands can be stored in the indexer to create a motion program. Communication between the host computer and the motor control is attained through a serial communication software via RS-232 interface.

When the linear positioning system is powered, the resolution of the motor must first be selected through the dip switches located on the motor control, and the ICL parameters must be initialized. The ICL parameters, such as the base speed, acceleration time, deceleration time, and motor resolution, are defined to perform the desired motion correctly. Motor units described in steps per revolution, encoder units given by encoder resolution per revolution, or user units usually given by rotational displacement in degrees per revolution must be selected to define the units in which distance will be determined. The carriage on the linear positioner can be moved sporadically given a finite number of steps or pulses to move or continuously until the motor control is given the command to stop. The initialization procedure for the crack detecting program utilizes the continuous motion of the carriage, while the tracking portion moves the laser sensor a finite number of pulses after receiving the offset distance from the laser sensor.

The hardware interface of the motor control allows the motor control to interface with the external devices, such as limit switches and encoder. To determine the position along the linear positioner, an encoder reading is obtained. Activation of the limit switches result in an error message that notifies the user that a limit switch has been reached.

4.2 Previous Approaches to Automated Pavement Crack Sensing

Various methods for detecting pavement cracks have been implemented to automate the highway maintenance task of crack sealing. Two different approaches for crack identification have been demonstrated on the ACSM through the use of the Vision

Sensing System and the Local Sensing System. The first approach utilizes the histogram, while the other approach, conceived by Debra Krulewich, a former graduate student, is statistical in nature.

4.2.1 Histogram Based Approach

Kirschke and Velinsky (1992) have presented a method for locating cracks in pavement using a histogram based machine vision approach. This approach locates cracks between 3.18 mm (0.125 in) to 25 mm (1 in) wide in both asphalt concrete (AC) and portland cement concrete (PCC) pavements. The VSS on the ACSM utilizes this approach to determine the presence and the orientation of the crack for purposes of path planning.

4.2.1.1 Methodology

A video image, with a pixel resolution of 1.59 mm (0.0625 in), is first taken of the pavement, and the image is divided into a grid as shown in Figure 4.4. The histogram based approach builds a histogram for each tile within the grid, where each tile is further broken down to 32 x 32 pixels. A histogram is a graphical representation of the number of pixels at a given intensity, displaying the concentration of pixels versus brightness in an image. Tiles that contain the image of a crack will have a higher distribution of darker pixels than tiles displaying no crack.
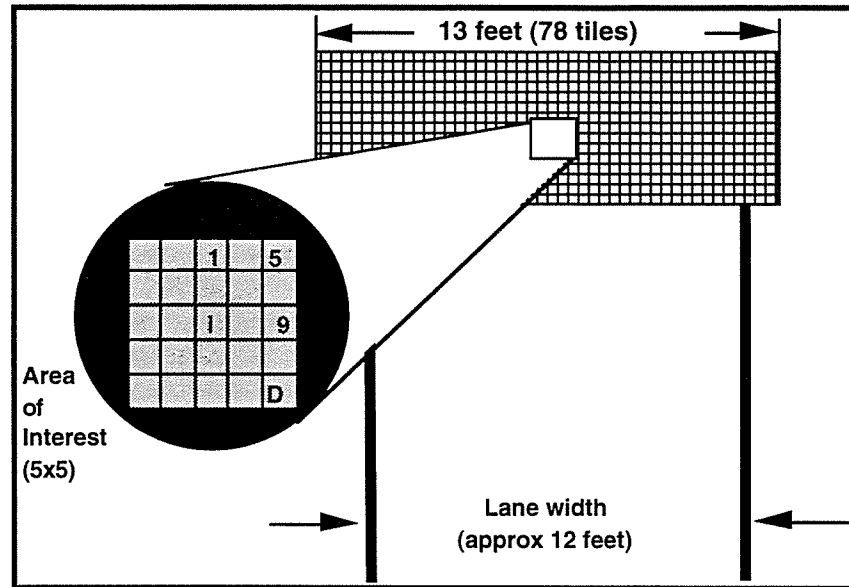
Figure 4.4 *Image of a Full Lane Width Broken Up into Tiles*

The statistical moment of each tile is then computed about the mode of the histogram and then compared to the statistical moment values of adjacent tiles to determine the presence of a crack. The mode of the histogram is the peak of the histogram, or the most frequently occurring pixel value in a given tile. The orientation of the crack is found by comparing a 5 x 5 grouping of tiles to the 16 possible directions stored in memory. Every tile in the image grid except for the tiles on the ends become the center tile in the 5 x 5 area of interest. For further clarification, refer to Kirschke and Velinsky (1992) and Velinsky (1993).

4.2.1.2 Shortcomings

This vision based crack detecting system has its disadvantages due to large memory requirements and the vision system's inherent limitation. An image of the

pavement is 25 cm x 370 cm (10 in x 12 ft), therefore the image grid requires enormous amounts of computer memory for storage. Because machine vision solely relies on pixel intensity, digitized images of pavement with oil spots, previously sealed pavement, or portions of pavement in the shade have been misdiagnosed as a crack.

4.2.2 Krulewich's Approach

To overcome the shortcomings of the histogram based approach for crack detection, a laser range finder based local sensor was used in conjunction to the VSS. The LSS verifies the existence of the identified crack to prevent any misinterpretations of the crack's presence. Krulewich and Velinsky (1992) examine another crack detection method which relies on statistical analysis of the initial data of a pavement sample obtained from the LSS and determines the presence of a crack using the results of the analysis.

4.2.2.1 Methodology

The laser sensor is placed over a section of pavement containing no crack, and a profile of the pavement is captured. The first twenty-five measurements of depth are used to extract the average depth along the acquired profile. The average depth is needed to calibrate the recursive high pass filter which the twenty-five depth values are passed through to calculate the tolerance of the pavement's surface roughness. The laser sensor is then placed over the section of pavement to be scanned. Figure 4.5 illustrates the

average depth and tolerance of the uncracked pavement surface overlaid over a section of

pavement with a crack to aid in the understanding of Krulewich's approach.
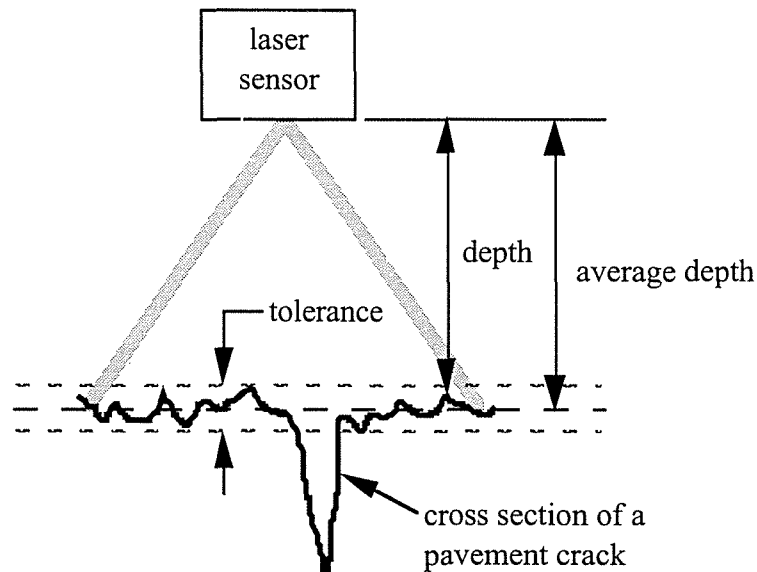
Figure 4.5 *Illustration of the Methodology Behind Krulewich's Approach*

The profile is captured and the depths are once again analyzed. The depth along

each point of the profile is compared to the depth of the previous point to determine

whether the starting edge of the crack has been found. If the current depth exceeds the

previous depth plus the tolerance, the first crack edge has been detected. After the

starting edge of the crack has been found, the other edge of the crack is encountered when

the present depth is less than the previous depth plus the tolerance. After detecting both

edges of the crack, the width is calculated to see if the crack is wide enough to be sealed.

If the width of the crack is not within the acceptable range of 3.18 mm (0.125 in) to 50

mm (2 in), the operator is notified. If none of the depths along the profile exceeds its

previous depth measurement plus the tolerance, then a crack is not present, and the

operator is notified.

4.2.2.2 Shortcomings

The effectiveness of Krulewich's approach relies greatly on the tolerance value that is initially obtained and requires the surface to be matte. If the tolerance value is too large, a crack is not detected, and if the tolerance is too small, normal surface deviations in rough pavement could be interpreted as a crack. The reflectivity of the surface is another source for erroneous results. Reflective particles that compose the pavement produce scattered bright pixels throughout the image, which become filtered into the data and yield erroneous results for average depth and tolerance.

4.3 The Crack Finding Approach for the TMRR Using the Gradient Method

Employing the same LaserVision sensor as the LSS on the ACSM, a more robust crack finding approach is needed to provide accurate and reliable crack position information for the success of the TMRR's operation. Previously, the LSS was used in conjunction with the VSS and therefore had prior knowledge of the whereabouts of the crack, however the TMRR, working autonomously, will not have this prior knowledge and will need to rely solely on the LaserVision sensor to detect the presence of the crack. Instead of implementing a statistical approach to process the pertinent information from the crack profile returned by the manufacturer's software, simple image processing techniques were examined. Of the many image analysis operations, enhancement or

detection of edges in an image is a commonly used operation. Edge enhancement, a form of feature extraction that reduces the image to display only its edges, will facilitate the extraction of crack edges from the profile.

Edge enhancement filters, particularly the gradient filter, has been examined. Edge enhancement filters typically manipulate the brightness of each pixel. However, MVS' software already extracts the brightest pixels from the CCD array, so the extracted pixels are the pixels that fall in line with the laser beam.

4.3.1 The Gradient Filter

The gradient filter, or shift and difference method, is commonly used for edge enhancement. This edge enhancement technique is based on the slope of pixel brightness occurring within a pixel group. If the slope is steep, the gradient will be large and if the slope is gradual, the gradient will be small. If there is no change in the surface roughness, the gradient is equal to zero. Because edges are sharp changes in brightness, the presence of an edge is characterized by large slopes.

Although the image captured by the camera is a two dimensional image, MVS' software reduces the image to a one dimensional image by considering only a single row from the image, which is made up of the brightest pixels in the CCD array. Therefore, the gradient or the first derivative of the extracted profile is defined as

$$\frac{\partial I_i}{\partial x} = \lim \frac{I_{i+1} - I_i}{\delta x} \qquad (4.1)$$

where I_i represents the ith pixel in a row of image I and δx represents the distance between adjacent pixels. The distance between adjacent pixels is the difference between the line numbers, which is equal to one. Thus, the gradient becomes the difference between the subsequent and present pixel.

4.3.2 Crack Detecting Algorithm for the TMRR

A crack detecting algorithm using the gradient approach minimizes processing time and is more robust than Krulewich's Approach. To reduce the processing time, acquiring a profile of a sample of pavement that is not cracked is eliminated. The laser sensor system captures a profile of the pavement, and the pixel values, that are obtained, are passed through a thresholding filter. Thresholding the data eliminates the problem associated with reflectivity of the pavement's surface. Knowing the range in which the valid pixel values lie, scattered bright pixels caused by the reflectivity of the pavement surface can be easily eliminated because they are preposterously too large or too small to fall within the valid range. If the first pixel in the image is not within the valid range, the average of all the pixel values is substituted in its place. While the remaining pixels do not fall within the acceptable range, the current pixel is replaced with the value of the preceding pixel. The data is then passed through a recursive high pass filter that was used in Krulewich's approach and then through the gradient filter. The maximum and minimum values are extracted along with their respective positions along the profile. If the maximum is greater than 3.0, then the first edge of the crack is detected and the trailing

edge is sought. Otherwise, a "no crack found" signal is sent to the operator. If a crack is detected, the width is calculated and compared to the acceptable range of crack widths. If the crack width valid, then the average of the two positions of the crack's edges is taken to obtain the location of the crack's center with respect to the center of the TMRR. Otherwise the crack width is not acceptable for sealing, and a warning is sent to the operator stating that the crack is too small or too large for sealing. The flow chart for the gradient approach for crack detection is displayed in Figure 4.6.

This crack detecting method is performed during the initialization and the main body of the main program. The crack sensing algorithm is executed continuously when the TMRR is initially placed onto the cracked pavement to locate the start of the crack. The laser sensor scans the pavement for the presence of a crack while the linear positioner moves the laser sensor from one end until the crack is detected. Once the crack is found, the crack sensing algorithm runs continuously to track a crack while the TMRR is in motion.

4.3.3 Crack Tracking

Tracking of the crack is done by constantly locating and updating the position of the center of the crack. The position of the crack with respect to the center of the laser sensor is given as an offset error. The offset is converted into motor pulses and sent to the linear positioner to place the laser sensor over the center of the crack.

Capture a profile of the pavement surface

Replace the pixel with the average of all the pixels ← NO — Is the first value in the pixel array within the acceptable range?

YES

Is the next pixel value within the acceptable range? — NO → Replace the pixel with the previous pixel

YES

Pass data through high pass recursive filter

Take the derivative or difference of the two consecutive pixels and store the results in the derivative array

Search and locate the maximum value in the derivative array

Search and locate the minimum value in the derivative array

Send warning: No crack found ← NO — Is the maximum greater than 3.0?

YES

Take the difference of the locations of the crack edges along the profile to obtain the width of the crack

Send warning: Crack is too large or too small to be sealed ← NO — Is the width of the crack between 1/8 to 2 in?

YES

Take the average of the locations of the crack edges along the profile to obtain the location of the crack's center

Send the position offset back to the computer

Figure 4.6 *Flow Chart of the Crack Detecting Algorithm for the TMRR*

To eliminate minute oscillations of the laser sensor caused by small changes in the location of the crack's center, a window was added into the program. If the distance in which the linear positioner was required to move was less than 6.4 mm (0.25 in) in either direction, then the linear positioner would keep the laser sensor stationary. The position of the laser sensor with respect to the center of the TMRR is returned by the rotary encoder mounted on the motor of the linear positioning system. The location of the crack with respect to the center of the TMRR is needed to place the router and the sealant applicator accurately over the detected crack.

The tracking operation is terminated when 50 "no crack found" signals are sent to the operator continuously. This signifies that there is not a crack in the pavement, hence, the end of the crack has been attained.

A copy of the crack finding program, which was written in C and compiled on Microsoft C version 7.0, can be found in Appendix E. To reduce the length of the program, functions were compiled separately and linked together to form a library. A copy of the functions and the program stored on the linear positioner's motor control as part of the initialization procedure are also included in Appendix E. This program calls several functions supplied with the LaserVision system, however they are already in the form of libraries, so the code for the supplied functions are not available for viewing.

The crack detecting program operates in real time on an industrial Pentium 90 MHz computer.

4.4 Summary

Because crack detection is the most important operation of the TMRR, the proper equipment was selected and a robust crack finding routine was desired. By examining the previous approaches in automating the crack detecting process, an alternative method was created to overcome the weaknesses of the preceding techniques. The TMRR's crack detecting algorithm utilizes simple image processing operations to yield shorter processing time than Krulewich's approach and to require less memory than the histogram approach. To evaluate the performance of the TMRR's crack detecting algorithm, tests need to be conducted. The results of the performance tests for the TMRR's crack detecting algorithm will be presented in the subsequent chapter.

CHAPTER 5: EXPERIMENTAL VERIFICATION

This chapter evaluates the performance and validity of the crack finding algorithm that the TMRR will be using to sense and track cracks in pavement. The first section will verify the effectiveness of the gradient method, and the second section will verify the crack tracking operation made possible through the integration of the linear positioning system and the local sensing system.

5.1 Experimental Verification for Crack Identification

To evaluate the performance of the crack detecting method that the TMRR will be utilizing, profiles of a groove or crack in different surfaces have been analyzed by applying the suggested method. The following profiles are representative of the many profiles that were taken for each surface to evaluate and obtain the proper parameters for the crack finding algorithm. All profiles were taken with the laser sensor maintained above the surface such that the bottom of the sensor was 8.26 cm (3.25 in) off the surface. The digitized image returned by the LaserVision System is inverted on the plots below due to the numbering sequence of the pixel value.

5.1.1 Examining Cracks in Pavement

To test the edge detecting performance of the gradient method, a simple indoor test was performed. The laser sensor was positioned so that the line of light produced by the sensor onto the surface was perpendicular to the direction of the groove. The profile

of the groove was captured, and the crack detecting algorithm was applied to the profile. The data obtained from the digitized image was filtered and processed using the gradient method to detect the edges of the groove. Figure 5.1 displays the superimposition of the resulting image onto the raw data. The two spikes clearly mark the edges of the routed groove. The spikes indicating the edges of the groove are much greater in magnitude than the spikes created by the deviations in the plywood's surface. Because the difference between the magnitudes of the smaller spikes and the larger spikes are so great, the larger spikes are easily extracted. The two upward pointing arrows on the chart indicate the locations of the edges of the rout after removing the phase shift created by the Butterworth Filter.

Profile of a 6.35 mm (0.25 in) Wide Groove in Plywood
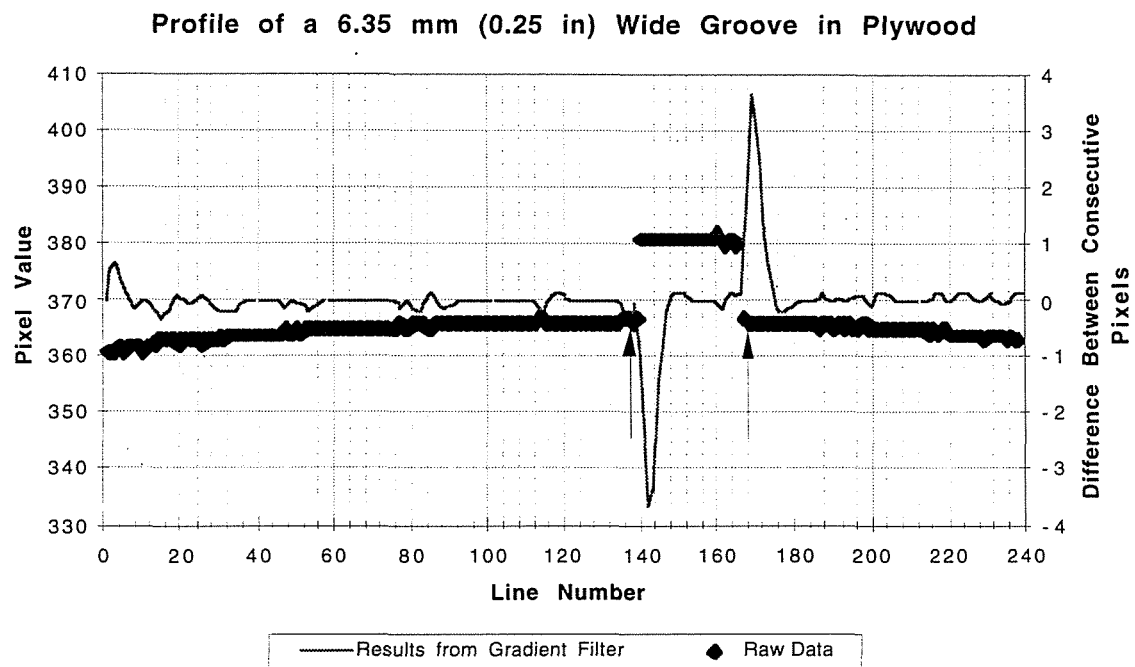
Figure 5.1 *Filtered Data Overlaid onto the Raw Profile of a Groove in Plywood*

Because the surface of the plywood is smoother than asphalt and the interior walls of a crack are never perfectly vertical, the need to apply the crack detecting algorithm to a more realistic model of a crack in a surface that was similar in roughness to pavement. Hence, the LaserVision System was transported to an asphalt parking lot to obtain a profile of an actual crack in asphalt pavement. Figure 5.2 displays the profile of the crack in asphalt pavement superimposed with the data obtained from the gradient filter. Once again, there are noticeably large spikes adjacent to each crack edge, and the edges of the crack returned by the algorithm are indicated by the upward pointing arrows.

Profile of a Crack in Asphalt Pavement
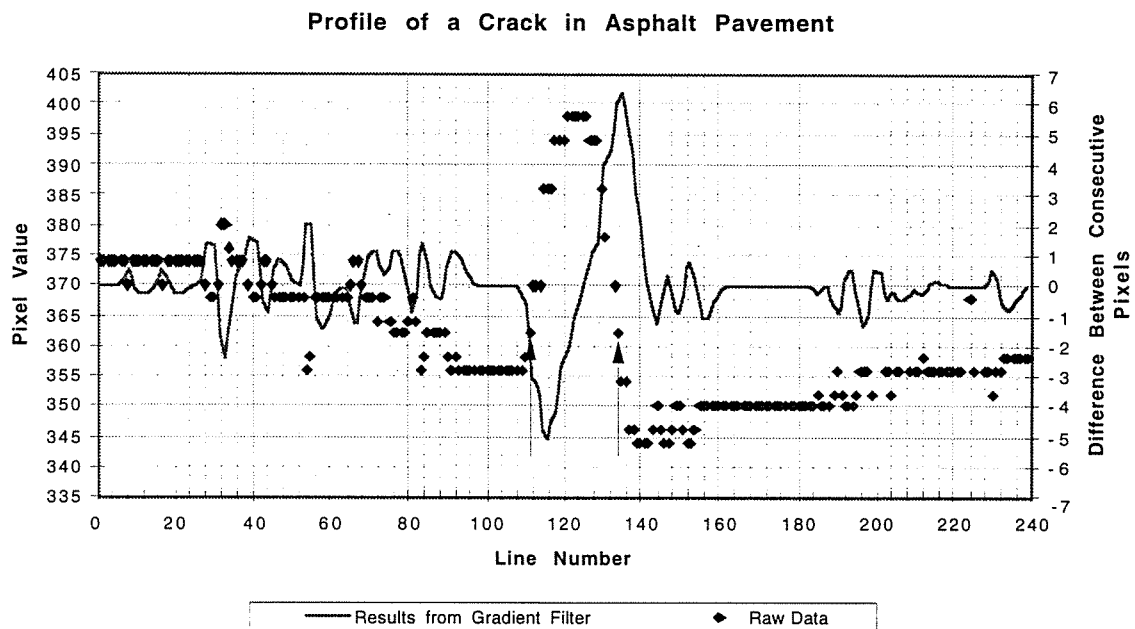
Figure 5.2 *Filtered Data Overlaid onto the Raw Profile of a Crack in Asphalt Pavement*

Because we are interested in sealing cracks in asphalt and concrete pavements, the crack detecting algorithm was also tested on concrete pavement. Figure 5.3 exhibits the profile of a crack in concrete pavement. Concrete pavements are smoother than asphalt

pavements and as a result reduces the number and magnitude of the smaller spikes that are generated by the gradient filter.

Profile of a Crack in Concrete Pavement
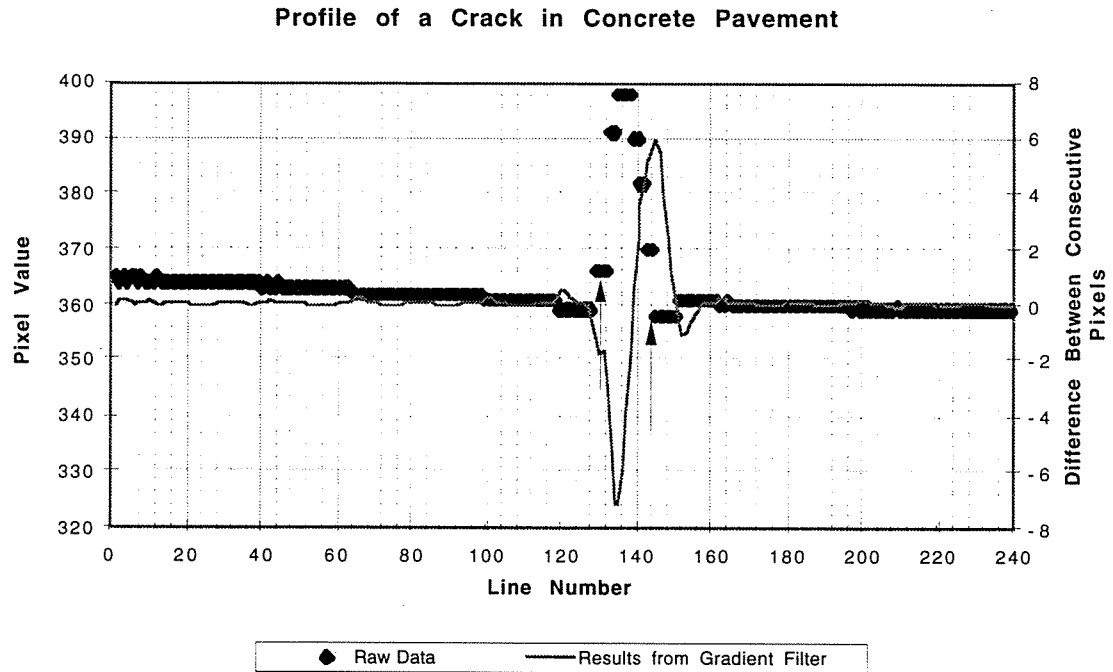
Figure 5.3 *Filtered Data Overlaid onto the Raw Profile of a Crack in Concrete Pavement*

The crack detecting algorithm has demonstrated its effectiveness, reliability, and robustness in detecting the edges of a crack in a rough surface. Once the edges of the crack are located, their locations are obtained by converting line number to the position along the profile in given in millimeters. This conversion function is provided by MVS. After the edge locations are given as positions along the profile, the center of the crack is computed by taking the average of the two positions and the width is given by the difference of the two positions.

5.1.2 Examining Pavements with No Crack

In order to form a criterion that would allow the program to differentiate between cracked and uncracked pavement, samples of pavement without any cracks were analyzed. Figure 5.4 shows a profile of an asphalt pavement surface that is not cracked. Although the spikes generated by the gradient filter are scattered along the profile, the magnitude of the largest spike is small compared to the spikes that are produced when a crack is present.

Profile of the Asphalt Pavement Surface with No Crack
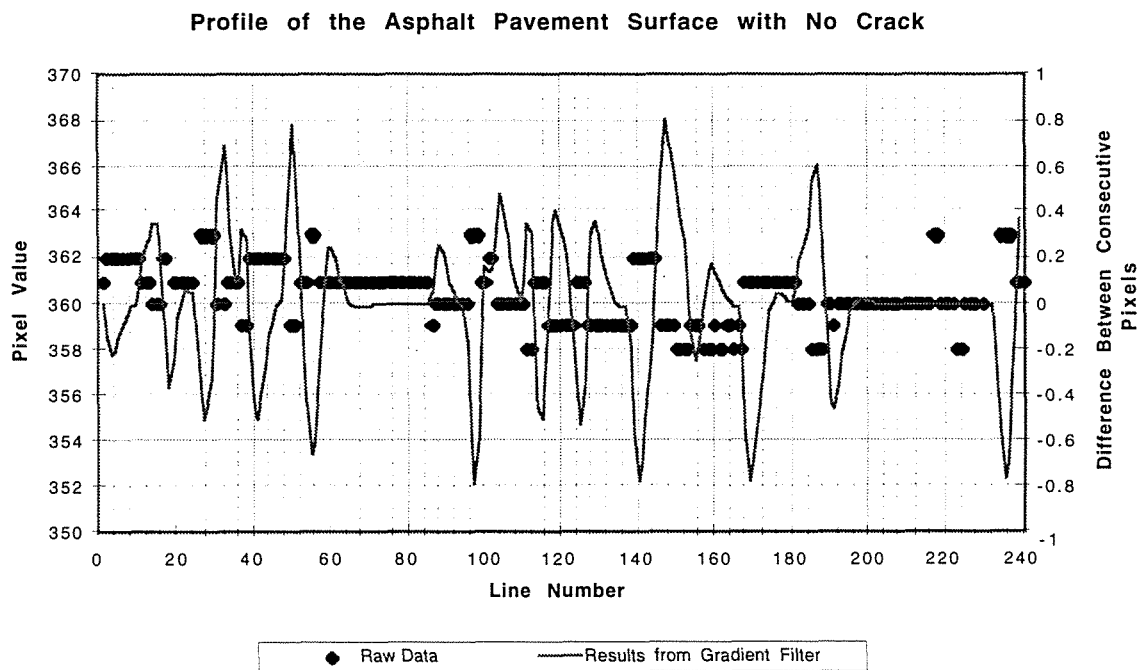
Figure 5.4 *Filtered Data Overlaid onto the Raw Profile of an Asphalt Pavement Surface that Does Not Contain a Crack*

Figure 5.5 displays the data obtained from the gradient filter overlaid onto the profile of a concrete pavement surface that does not contain a crack. Notice that the magnitudes of the spikes produced by the gradient filter are smaller than the magnitudes of the spikes generated by the asphalt pavement data due to its smoother surface.

Profile of a Concrete Pavement Surface with No Crack
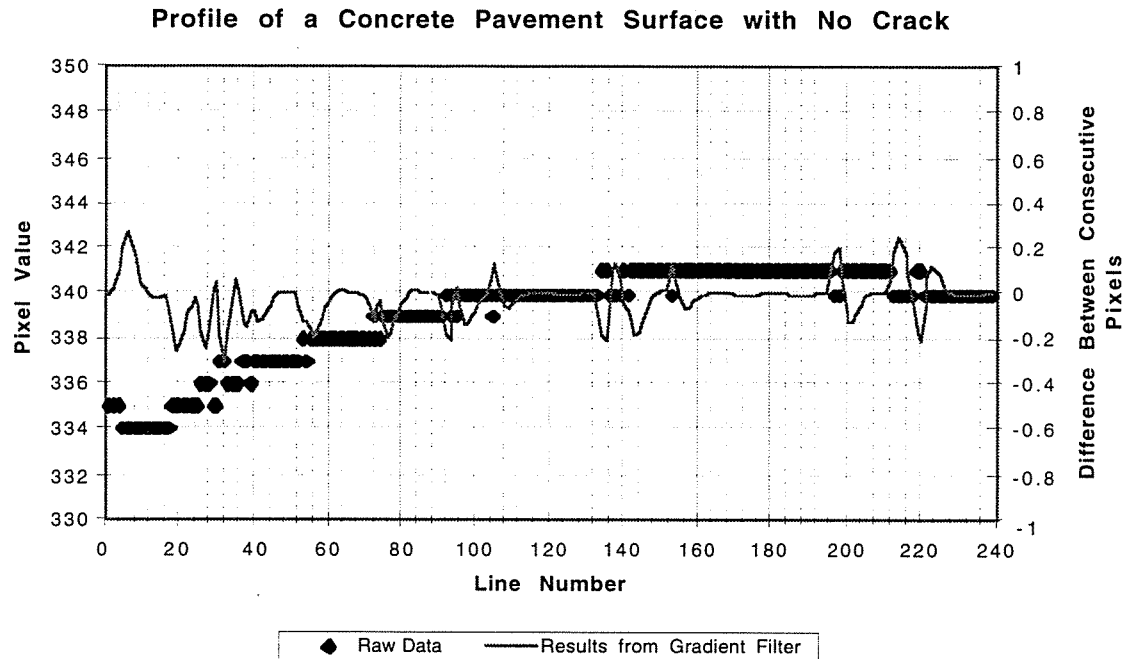
Figure 5.5 *Filtered Data Overlaid onto the Raw Profile of a Concrete Pavement Surface that Does Not Contain a Crack*

5.1.3 Crack Determining Criterion

Upon comparing the largest negative spike produced by the gradient filter of all the samples taken, the largest negative spike with a magnitude of 3 or larger were characteristic of a crack edge. If the magnitude of the largest negative spike had a value of 3 or greater, then that spike would denote the starting edge of the crack. Once the starting edge is located, the largest positive spike would be sought, indicating that the other edge of the crack has been located. Otherwise it would be assumed that the closing edge of the crack was not within the laser sensor's field of view.

5.1.4 Valid Pixel Range for the Thresholding Filter

The profiles of pavements and plywood were used to determine the valid pixel range for the thresholding filter. Upon comparing the pixel values that described the surface for the two types of pavement and the plywood, it is reasonable to assume that a valid pixel range is described by pixel values of 330 and higher. Small pixel values, such as those less than 50, are the products of reflective particles that compose the surface, and should therefore be ignored.

5.2 Experimental Verification for the Tracking Operation

Two tests were performed to evaluate the tracking performance of the integrated crack detection system. The first test evaluates the response of each crack detection system separately while performing jointly. The response of the linear positioning system is rated by comparing the distance moved by the linear positioner to the offset obtained by the laser sensor. The response of the local sensing system is noted by comparing the offset returned by the crack finding algorithm to the known offset. The second test assesses the overall tracking performance of the TMRR by having the crack detection system track a crack while the TMRR is in motion. The position of the crack's center returned by the main program is compared to the actual position of the crack with respect to the workspace.

5.2.1 Initial Tracking Test

A test was devised to verify the performance of both the linear positioning and local sensing systems while working together. For this test, the laser sensor was detached from the linear positioner, while retaining the same height of 8.26 cm (3.25 in) off of the ground. A rod is placed under the laser sensor, such that the offset of the rod's centroidal axis from the midpoint of the laser line would be a fixed distance, and the rod is oriented perpendicular to the line of laser light. Hence, while the laser sensor observed the rod's offset, the linear positioner moved the carriage accordingly until 40 samples were taken. Prior to commencing the test, the carriage on the linear positioner was moved to one end of its full stroke length to allow for the continuous displacement of the carriage towards the opposite end. The offset, obtained by the laser sensor, and position of the carriage on the linear positioner were recorded after the offset was converted to a "move" command, and the carriage on the linear positioner was displaced.

The crack detecting algorithm was slightly modified for this test to detect a protrusion instead of a recession in the surface. Rather than identifying the edges of a crack, the program will detect the edges of the rod. The method used to locate the object is the same, however, the valid pixel range has been modified, and the algorithm will first be searching for a positive spike and then a negative spike once the data has been processed by the gradient filter.

A rod with a diameter of 15.9 mm (0.675 in) was placed on the plywood such that its centroidal axis was displaced 9.53 mm (0.375 in) from the midpoint of the laser line.

The offset obtained by the laser sensor and the actual distance displaced by the carriage on the linear positioner were plotted for the 40 samples as shown in Figure 5.6.
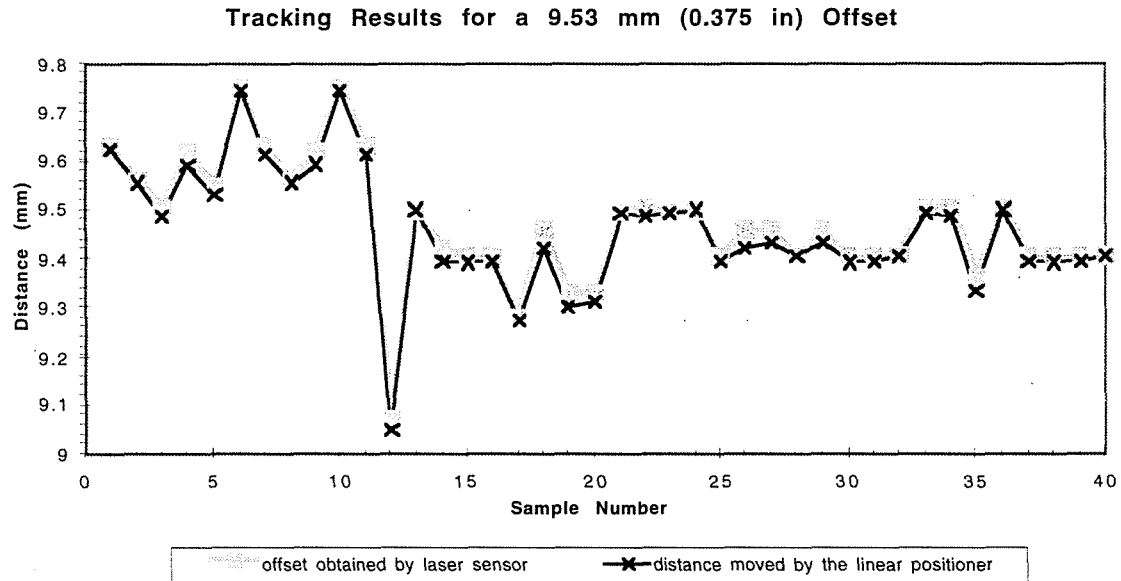
Figure 5.6 *Tracking Results of the Integration of the LaserVision and Linear Positioning Systems*

The data for the plot in Figure 5.6 are tabulated in Table 5.1 in addition to the calculated offset and tracking errors. The offset error was obtained by taking the difference between the offset obtained by the laser sensor and the known offset of 9.53 mm (0.375 in). The average offset error obtained for this test was 0.105 mm (0.00413 in) with the maximum offset error being 0.450 mm (0.0177 in) and the minimum offset error being 0.0203 mm (0.000799 in).

Sample Number	Offset Obtained by Laser Sensor (mm)	Distance Moved by Linear Positioner (mm)	Offset Error (mm)	Tracking Error (mm)
1	9.62914	9.6266	0.10414	0.00254
2	9.55802	9.55675	0.03302	0.00127
3	9.50468	9.4869	0.02032	0.01778
4	9.61898	9.5885	0.09398	0.03048
5	9.54786	9.53135	0.02286	0.01651
6	9.7536	9.74725	0.2286	0.00635
7	9.62914	9.6139	0.10414	0.01524
8	9.55802	9.55675	0.03302	0.00127
9	9.61898	9.59485	0.09398	0.02413
10	9.7536	9.74725	0.2286	0.00635
11	9.62914	9.6139	0.10414	0.01524
12	9.07542	9.04875	0.44958	0.02667
13	9.49452	9.4996	0.03048	0.00508
14	9.4234	9.398	0.1016	0.0254
15	9.40308	9.39165	0.12192	0.01143
16	9.40308	9.398	0.12192	0.00508
17	9.28116	9.27735	0.24384	0.00381
18	9.45896	9.4234	0.06604	0.03556
19	9.33196	9.30275	0.19304	0.02921
20	9.33196	9.3091	0.19304	0.02286
21	9.49452	9.49325	0.03048	0.00127
22	9.50468	9.4869	0.02032	0.01778
23	9.49452	9.49325	0.03048	0.00127
24	9.49452	9.4996	0.03048	0.00508
25	9.40308	9.398	0.12192	0.00508
26	9.45896	9.4234	0.06604	0.03556
27	9.45896	9.42975	0.06604	0.02921
28	9.40308	9.40435	0.12192	0.00127
29	9.45896	9.42975	0.06604	0.02921
30	9.40308	9.39165	0.12192	0.01143
31	9.40308	9.398	0.12192	0.00508
32	9.40308	9.40435	0.12192	0.00127
33	9.50468	9.49325	0.02032	0.01143
34	9.50468	9.4869	0.02032	0.01778
35	9.36244	9.3345	0.16256	0.02794
36	9.50468	9.4996	0.02032	0.00508
37	9.40308	9.398	0.12192	0.00508
38	9.40308	9.39165	0.12192	0.01143
39	9.40308	9.398	0.12192	0.00508
40	9.40308	9.40435	0.12192	0.00127
		Average Error:	0.1054735	0.0132715
		Maximum Error:	0.44958	0.03556
		Minimum Error:	0.02032	0.00127

Table 5.1 *Tracking Data for the 9.53 mm (0.375 in) Offset Test*

The tracking error was determined by taking the absolute value of the difference between the offset obtained by the laser sensor and the distance moved by the linear positioner. The average tracking error was 0.0133 mm (0.000524 in). The maximum and minimum tracking errors were 0.356 mm (0.0140 in) and 0.00127 mm (0.00005 in) respectively.

5.2.2 Crack Tracking Test

With the crack detection mounted onto the TMRR and the crack detecting algorithm incorporated into the main program, the crack tracking ability of the TMRR was evaluated. The position of the center of the crack with respect to the workspace was saved in a file as the TMRR tracked a meandering routed crack in plywood while in motion. The location of the crack's center was retrieved from the file and plotted against the actual location of the crack's center, which was obtained by manually drawing a grid on the workspace. The resulting plot is displayed in Figure 5.7.

The line representing the location of the crack's center returned by the main program conforms to the actual crack location. The computed crack location slightly differs from the actual crack location due to some error. The error is the sum of human error and errors produced by the crack detection system and while computing the location of the TMRR with respect to the workspace using the CETs.

TMRR Tracking a Routed Crack in Plywood
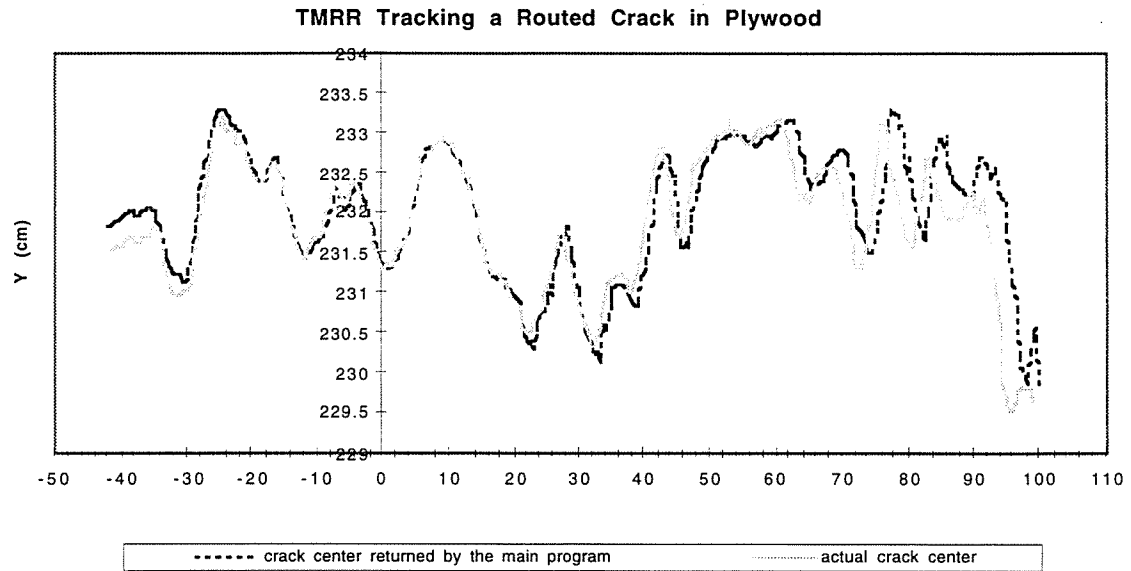
Figure 5.7 Crack *Tracking Results of the TMRR*

5.3 Summary

The gradient method has proven to be practical in edge detection. The crack finding algorithm utilizes the gradient method to detect the edges of a crack and from that information, determine the crack's position with respect to the center of the TMRR. The crack finding algorithm has demonstrated its efficacy in locating a crack in both asphalt and concrete pavements.

The range of offset and tracking errors resulting from the initial tracking test verifies the precision of the LaserVision and linear positioning systems. The accuracy and precision of the laser sensor and the linear positioner were proven by recording errors that were less than 0.508 mm (0.002 in). The magnitude and range of the tracking errors were smaller than the offset errors. Although the location of the crack, that is returned by

the crack finding algorithm, maybe slightly displaced from the actual position, the error is trivial in light of the crack sealing operation requirements.

The results of the crack tracking test demonstrates the TMRR's ability to track a routed crack in plywood. Although there is an offset between the computed and actual crack locations, the error is the sum of human, mechanical, and computational errors. From the initial tracking test, we have seen that the errors generated by the crack detection system were relatively small, therefore the majority of the error is due to human error and errors generated upon computing the location of the TMRR with respect to the workspace.

Combining the effectiveness of the crack detecting algorithm and the accuracy of the LaserVision and linear positioning systems, the crack sensing system has proven itself to be effectual and capable of locating and tracking cracks.

CHAPTER 6: CONCLUSIONS AND RECOMMENDATIONS

6.1 Conclusions

The progression towards automating highway maintenance operations has led to the development of a several crack sealing machines by AHMCT. The first two crack sealing machines were the longitudinal crack sealing machine and the general crack sealing machine. Combining the idea behind each of these crack sealing machines and applying the TMR concept led to the evolution of the TMRR.

The TMRR executes three basic tasks to complete the crack sealing operation. Similar to the manual crack sealing operation currently performed by highway maintenance workers, the TMRR will locate, rout, and seal cracks. The crack finding algorithm is part of the task planning for the overall control scheme of the TMRR.

Crack sensing is essential for a successful automated crack sealing procedure. Flawless routing and sealing of cracks depends on the accurate location of the crack with respect to the center of the TMRR. The center of the crack is located by applying an effective crack detecting algorithm to the data acquired by the laser sensor, while its position with respect to the center of the TMRR is determined by the encoder feedback of the linear positioning system. This thesis has developed an algorithm for the TMRR to detect a crack in pavement using the local sensing system in conjunction with the linear positioning system. The crack finding algorithm automatically identifies the crack using the gradient method to first locate the edges of the crack and then determines the location of the center of the crack with respect to the center of the TMRR. Then the crack's

position with respect to the center of the TMRR is obtained by rotary encoder mounted onto the motor of the linear positioner and fedback to the computer.

The results of the edge detecting test evaluating the performance of the gradient method validates the use of the gradient method as a part of the crack finding algorithm on both asphalt and concrete pavements. The gradient filter produces distinct spikes adjacent to the edges of the crack, which expedites edge detection. Using the crack finding algorithm with some minor changes for the tracking test demonstrates the practicality of the gradient method for locating objects on a surface when using a laser sensor.

The results of the crack tracking test illustrates the . The offset error between the computed and actual crack locations is the sum of human, mechanical, and computational errors.

The trivial errors, generated by the initial tracking test, indicate the accuracy of the equipment that constitute the crack sensing system. The errors that were produced were less than 0.508 mm (0.002 in), which exceeds the accuracy that we had expected. The preciseness of the LaserVision and linear positioning systems coupled with the competence of the crack finding algorithm presents an effective means for tracking cracks in pavement.

6.2 Recommendations

The crack tracking task of the TMRR was the first task to be tested. The routing and the crack sealing operations have yet to be performed. Thus, the performance of the crack tracking system has not been under actual environmental conditions of the crack

sealing operation. Because we do not know how much of the vibrations generated by the router will be isolated by the air springs, we have mounted the laser sensor 8.26 cm (3.25 in) off of the ground. If the vibrations are well damped, then we could increase the height of the laser sensor to about 10 cm (4 in), which in turn would increase our field of view also. However, if the vibrations are not well isolated, the laser sensor may need to be lowered. In any case, altering the height at which the laser sensor needs to be placed above the ground will generate distorted profiles which will yield erroneous displacement measurements. Therefore, recalibration of the laser sensor, at the height at which it will be operating at, will be necessary to obtain the proper calibration table to yield accurate measurements when converting from camera coordinates to the user's coordinates. Additional modifications to the calibration file may be necessary if the profile is skewed. The profile may need to be translated and/or rotated for an accurate transformation from camera to user coordinates.

Bundling the electrical cables that run from the computer on board the support vehicle to the TMRR may induce noisy signals if the cables are sensitive to electrical noise. A common producer of electrical noise is electromagnetic interference (EMI). Noisy signals usually produce faulty results and may possibly cause the computer to crash. Properly grounding all electrical equipment and components and adequate shielding of the cables can reduce noisy signals.

The laser sensor will be subjected to an extremely dusty environment due to the routing operation. Although the laser sensor is enclosed within a robust casing, added precautions have been taken by designing an enclosure for the LSS. The LSS is enclosed

in a thin wall aluminum tube with a rubber skirt wrapped around the lower edge to keep out the dust and debris from harming and/or disrupting the LSS. If however, the router generates an excessive amount of dust, and the dust accumulates within the enclosure, the profile of the crack cannot be obtained because the cloud of dust will obstruct the camera's view of the laser line. To resolve this problem, the enclosure could be pressurized so that air is constantly flowing through the enclosure, forcing out any dust that has accumulated within the cylindrical enclosure.

Because the router will be upmilling, we have assumed that the cutters will be removing the fragmented pavement particles from the routed channel. However, if an abundant amount of debris remains in the routed crack, the crack will need to be blown out prior to sealing. The crack will be poorly sealed if there is too much debris within and surrounding the crack because the sealant will adhere to the rubble and not the pavement.

The crack detecting algorithm, as the main program that controls the TMRR, is compiled as a DOS executable program. To make the program more user friendly the program could be written for a Windows environment. Because we plan to expand the TMR concept to apply to multiple TMR systems, we would eventually upgrade the program to work in a Windows multi-tasking environment so that each TMR system could be controlled separately.

APPENDIX A

Manufacturer's Specifications for the LaserVision Sensor

Resolution and Accuracy

The Laser Vision camera is calibrated to work in the above mentioned optimum area. All accuracy and resolution specifications are specified for this area.

	Horizontal		Vertical
Speed images/sec	60	30	
Resolution:	0.005" 0.125mm	0.0025" 0.064mm	0.006" 0.15mm
Accuracy position:	0.006" 0.15mm	0.003" 0.076mm	0.008" 0.2mm
Accuracy mismatch:			0.002" 0.05mm
Accuracy gap:	0.012" 0.3mm	0.006" 0.15mm	

Camera Bracket

MVS-30 Weld pool
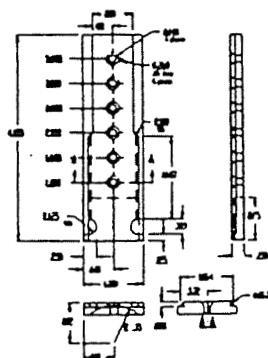
Mounting

The LaserVision sensor is mounted on the torch using the camera bracket supplied. This precision machined part should be installed without any warping on a custom machined and insulated bracket mounted on the welding torch. Mounting should ensure flexibility of vertical or lateral adjustment. A 5° sensor tilt towards the torch tip is recommended. The distance to torch tip should be as short as possible, but at least 0.5" longer than the longest expected tack weld.

Applications

The MVS-30 LaserVision sensor is a medium resolution sensor specifically designed for both tracking and inspection robotic applications. The elongated field of view helps in the initial part location, as well as the weld pool observation. It is best used for V-grooved butt joints, large lap joints and fillet joints. The maximum lap joint height is about 1" or 25mm. MVS-30 sensor is designed for MIG, subarc, plasma and fluxcore with welding currents up to 900A.
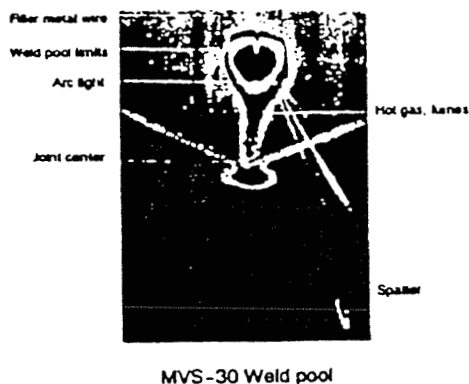
Specifications

Speed.   60 images per second - RS170
         50 images per second - CCIR standard

Cooling liquid 1/4 US gallon (1l) per minute (air
         cooling for subarc and currents up to 50 A)

Air.   0.11 CFM (3l per minute)

Weight: 9oz (250g)

MVS Modular Vision Systems Inc., 3195 De Miniac, Montreal, Canada, H4S-1S9, (514)-333-0140, FAX (514)-333-8636

LaserVision Sensor MVS-30 Specifications

General

LaserVision is a new generation of highly *reliable* laser range (profile) sensors with *no moving parts*, specifically designed for welding and sealant dispensing applications. It is the first really *affordable* vision based sensor, provided *high processing speed and reliable tracking* with more than adequate information for statistical process control and improved parameter control. At the same time, LaserVision is *simple to use and rugged* enough to provide trouble free service in any welding or other hostile industrial environment. A unique patented[1] design allows for over 200 *hours of maintenance free operation* under extreme spatter conditions (900A fluxcore). The output from the sensor is a common TV signal, allowing the images to be recorded for the Quality Assurance on a standard VCR.

Principle of Operation

The LaserVision sensor uses a laser light projected in a plane approximately perpendicular to the observed part. The cross section of the laser plane of light and the part produces a bright line. When this line is observed by a CCD camera at an angle (20° to 30°) it shows the surface features.

A dedicated vision processor board LPB-200 extracts the surface profile of 60 times per second – even under extreme arc light and spatter conditions. The relative distance of the surface points under the sensor is then calculated (by triangulation) and features of the profile, such as joint position and geometry, are extracted and measured.

Field of View

The field of view is trapezoidal in shape (see drawing) due to the angle of observation of the laser plane. An important feature of this approach is that a straight line remains a straight line, but angles are not preserved. This geometry allows for all the tracking algorithms to be performed in the camera space. A simple set of equations, with eight coefficients obtained by the LaserVision camera calibration procedure, describes the camera space and all the range points can be easily calibrated. The shaded area is the optimum working area for the LaserVision camera where resolution is highest, focus for both the laser line and the camera is optimal and distortion of the optics is minimal.

MVS-30 Outline and Optics
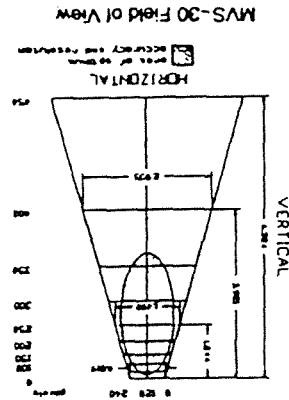
MVS-30 Field of View

3 LASERVISION SENSOR AND PROCESSOR

3.1 Introduction
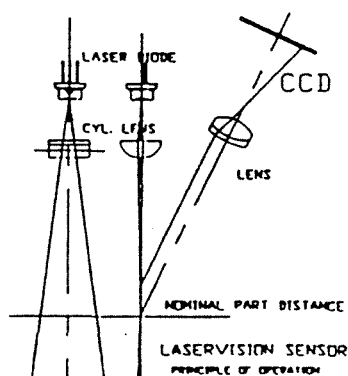
Figure 3.1.1 LaserVision Sensor
Optics and Principle of Operation

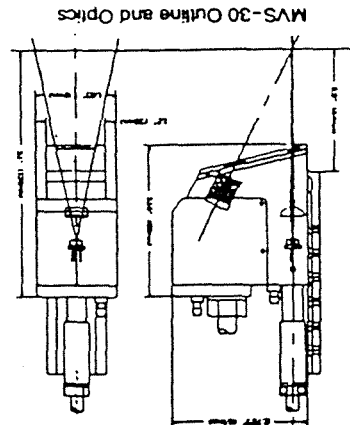The LaserVision sensor is a range type sensor. It uses a laser light projected in a plane approximately perpendicular to the observed joint. The cross section of the laser plane of light and the part produces a bright line. When this line is observed by a CCD camera at an angle (20° to 30°) it shows the surface features. A dedicated vision processor board LPB-200 extracts this *profile* of the surface 60 times per second even under extreme arc light and spatter conditions. A relative distance of the surface points under the sensor is then calculated (by triangulation) and features of the profile, as joint position and geometry, are calculated by an array processor SKY-320 and the AT-PC compatible computer. Each calculated joint position point is further verified, filtered and stored into the memory as a *trajectory queue*. This point is then output at appropriate moment to the positioning subsystem when the torch arrives at the position where this particular joint position is measured (see the Motion Control section of this manual).

3.2 The LaserVision Sensor

Warning: Please read LASER SAFETY INFORMATION! A serious eye injury can result if the laser safety is not respected.

The LaserVision sensor consists of a CCD camera (a solid state TV camera) and a semiconductor laser. A pinhole, lens and filter combination serves as the objective for the CCD camera. A cylindrical lens is used to focus the laser beam into a plane of light. The beam is further restricted by the slot on the sliding protective plate. Small glass windows are used behind the slots and in front of the lens in order to further protect the lenses from spatter and metal fumes.

The entire camera is pressurized to prevent welding fumes from entering. Pressure is relieved through both the laser slot and the pinhole. In order not to disturb a gas shield around the torch the direction of the blown gas is away from the weld pool and the amount of the gas used is minimal (3.5 litres, or slightly less than one US gallon per minute). A clean pressurized air or inert gas should be used. If shop compressed air is used a reliable water, oil and dust filter should be installed in the air line. 1/8" barbed connectors are used for air and cooling water connection, suitable for 1/8" PVC tubing. The connector is rated for 150 psi of pressure when proper tubing is used.

Water cooling is mandatory for open arc currents of more than 50A. Air cooling can also be used for applications of less than 50A on open arc or a subarc system with currents up to 200-300A .

The interference filters are:

- 30nm bandwidth for TIG arc up to 100A and wire feeder in front; up 50A TIG without wire feeder in front and subarc applications.

- 10nm bandwidth for higher current TIG, plasma, MIG and flux core wire.

- 5nm bandwidth for some very bright arcs, usually plasma, with use of a fibre optic laser.

In case of 10nm filter bandwidth chilled (and heated in case of low operating temperatures) water is required to maintain a precise operating temperature for the laser ($\pm3^{\circ}$C).

For more information consult the LaserVision sensor data sheets.

3.3 LaserVision Sensor Control and Processing

The camera video and synchronization signals are led via the camera power supply to the LaserVision Processing Board LPB-200 (200-SYS-01). The camera power supply is factory adjusted. If required, please refer to the CCD camera and power supply information included.

The laser intensity is also controlled by the same processor board via signal IPUL (ILIN in earlier versions). A Laser Filter Board LFB-265 (265-SYS-02) provides optical isolation for the laser intensity control signal and a dedicated *floating* laser power supply connections. The laser control signal is a pulse with modulated signal with 60Hz base frequency. Maximum intensity and linearity of the control is adjusted by the potentiometer P1.

The laser power supply +5.25V and -12.0V is switched by the relay R1 by the interconnection board IB-240. The IB-240 board enables the laser only if:

- the EMERGENCY STOP is not pressed,

- there is no ALARM condition (watch-dog timer) and

- the LASER push button is engaged.

The signal received by the LPB-200 board as well as the processed profiles and tracking cursors can be observed on a profile monitor led by the LPB-200.

For more information on the LaserVision Processing Board see the LaserVision Profile Processing Board - Technical Description

For the maintenance consult the LaserVision Camera Maintenance section.

4 LASERVISION PROFILE PROCESSING BOARD LPB-200

4.1 Introduction

MVS LaserVision Profile Board (LPB) is an image processing board specifically designed for extracting profiles of objects using a structured light and CCD camera. These profiles are generated by projecting a laser line on an object and observing it at an angle with a standard CCD video camera. Digital filtering techniques are used in order to ensure reliable operation in a high noise environment (i.e. arc welding) and suppress reflection artifacts.

LPB plugs into a single slot of an IBM-AT compatible computer.

4.2 LPB Main Features

The LPB can be used either alone as the only vision module in the system or with additional modules for increased performance, such as the DSP board with the Texas Instrument DSP processor TMS32010. A separate output port is provided for the transfer of profile data to the DSP board.

The principal features of the LPB module:

* Camera input, digitized at 8 bit per pixel.

* High resolution of 512 pixels per line standard.

* Highly stable digital phase locked loop synchronization of the internal pixel clock to the horizontal sync signal. A non cumulative jitter is less than +/- 12% of the pixel clock period, allowing for sub-pixel measurement accuracy. A reliable operation is achieved even with the standard VCR.

* Two groups of 2Kx8 (8Kx8 optional) bit input Look-up Tables, one group for processing and other for histogram.

* Monitor output with 8Kx8 (32kx8 optional) bit output Look-up Tables.

* Histogram circuit for 256 possible levels operating either on entire frame or area of interest window.

* Real time digital filter for the accurate feature extraction (profile)

* Profile extractor stores x, y coordinates and intensities of the most probable line points into 2Kx16 (8Kx16 optional) profile memory capable to contain 4 (16) profile vectors of 240 coordinates. This memory is accessible either to the AT host or a separate DSP processor board via a DSP output port.

* Eight selectable Area of Interest windows, easily movable around the picture area by specifying only the X-Y offset coordinates. Both histogram and profile extractor can be set to work only within this area of interest window.

- Flexible RAM based video clock and cursor generation allows for easy synchronization with wide range of standard and nonstandard video inputs.

- Capability to display "raw" profile or other intermediate results by simply loading pixel coordinate for each line into FIFO circuit.

- Laser intensity control output at 8 bit resolution.

4.3 Functional Overview

Main functions of the LaserVision Profile Board are shown on the Block Diagram (LPB.DWG).

The LaserVision camera is connected directly to the LPB via the provided cable. External synchronization is normally used, but there is a provision to use internal sync extraction from the video signal (VCR use) with some sacrifice in vertical positioning accuracy. Generation of internal synchronization signals is RAM based and allows for nonstandard video signals.

Initialization program supplied with the LPB loads necessary values for the RS-170 standard (North American B&W video) or CCIR standard (European) depending on type of camera. Same memory also serves for generation of two independent cursors.

The video signal coming from the camera is first conditioned then digitized to 8 bit accuracy. Two sets of look-up tables are provided, one for the digital filter and other for the histogram circuit. This allows entirely independent operation of the histogram circuit.

The digital filter is optimized for both noise suppression and laser line signal extraction. The laser line signals are enhanced and all other noise signals as ambient light are attenuated.

The digital filter circuit is followed by the profile extraction circuit. This circuit selects the peak of the laser line signal for each active video line and stores the result into Profile Memory during the horizontal blanking interval.

Results stored in the Profile Memory are accessible for further processing either by the host computer or via the DSP Output Port by the DSP board.

In order to further improve the noise immunity of the processing the LPB features the Area of Interest Window. Up to 8 different windows can be stored into window memory. Windows are selected through the control registers and they can easily be moved around the active video frame via X-Y offset registers. Both Histogram and Peak detector circuits can be set to operate only within the window and to ignore areas outside the window.

Typically the window shape is selected to closely match the expected joint profile. Once the joint profile is recognized and tracked, the window is set to closely follow the joint profile. Thus any noise outside the window of interest is automatically rejected. The described windowing technique also improves rejection of reflection artifacts.

The calculated profile or results of other intermediate calculations can be displayed on the monitor via the FIFO (first in first out) circuit. The FIFO has depth of 512 9 bit words, and it can be accessed via a single port. Total of 512 accesses fills the FIFO memory. When activated, the content of the FIFO memory is read synchronously with every active video line starting from the "zero" location. A single dot is output to the screen for every of 480 active lines at the pixel position equal to the address value stored into the corresponding FIFO location. The same output is automatically replayed every video frame without further program intervention.

The Output Look-Up Table circuit assigns gray levels to the digital filter results and the filtering operation can be observed in real time on the monitor. Windows, profile (FIFO) and cursors are displayed as bright overlay.

4.4 Specifications

Camera Input:	Video: 1Vpp
	Sync: TTL compatible, Standard RS170 or CCIR, other standards can be programmed.
Monitor Output:	RS170 or CCIR Composite sync.
Digitization:	Rate: 9.8304 Mhz standard
	Resolution: 8 bits.
	Jitter: less than +/- 12% of pixel width. Non cumulative.
Processor:	IBM-AT compatible, up to 8 Mhz, bus speed, requires 64K memory mapped space.
Output Port:	16 bit data, TTL compatible handshake control, up to 10 Mhz transfer rate, SKY320 compatible.
Measurements:	Width resolution: 240 points at 60 images per second, 480 points at 30 images per second, RS170 standard, or 256 points at 50 images per second, 512 points at 25 images per second, CCIR standard.
	Height resolution: 512 points.
	Nonlinear field of view due to triangulation technique used, however straight lines in actual space remain straight lines in transformed (camera) space. Calibration can be applied on final results only (i.e. after segmentation).

APPENDIX B

Optical Triangulation

Optical triangulation is a method of determining displacement of a diffusive surface. Because the LaserVision system by MVS projects a line of laser light onto the surface, the line-of-light system will be described. The line-of-light system in Figure B.1 is capable of acquiring two-dimensional measurements in the x and z directions. Figure B.1 displays a line of laser light projected onto a plate with a groove of width Δx and depth of Δz.
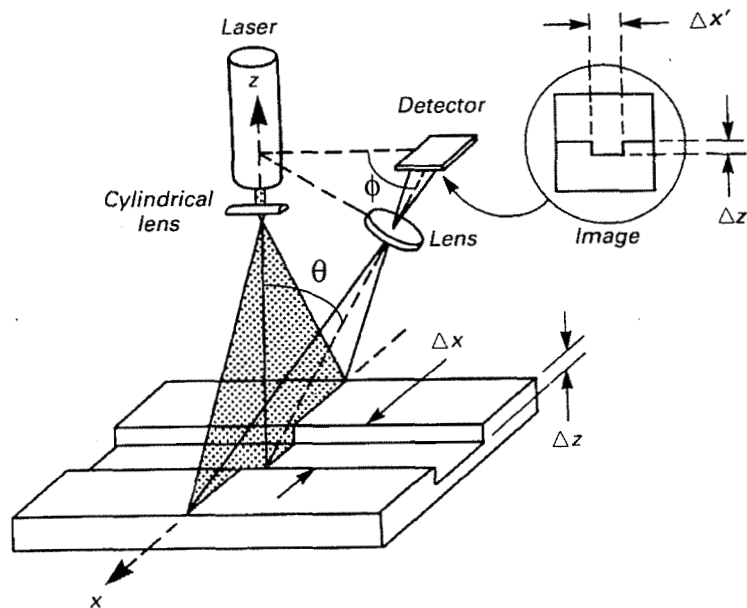
Figure B.1 *Diagram of Optical Triangulation of a Line-of-Light System Taken from Luxon et al (1992)*

A cylindrical lens is used to expand the laser light in one direction and not the other. The scattered light from the surface is focused onto a linear position detector

through a converging lens. The displacements $\Delta x'$ and $\Delta z'$ on the projected image is then used to determine the actual displacements Δx and Δz.

Using trigonometry, θ and ϕ are related to the o and i by

$$l = o\tan\theta = i\tan\phi. \tag{B.1}$$

Knowing the distances i and o and the angle θ, ϕ can be found using

$$\tan\phi = \frac{o}{i}\tan\theta. \tag{B.2}$$

The ratio $-i/o$ is the magnification, so the previous equation can be rewritten as

$$\tan\phi = \frac{1}{|m|}\tan\theta. \tag{B.3}$$

Therefore the displacement $\Delta z'$ of the projected image is found by

$$\Delta z' = m\frac{\sin\theta}{\sin\phi}\Delta z \tag{B.4}$$

and the displacement $\Delta x'$ is determined using the principle of transverse magnification given by the relationship

$$\Delta x' = m\Delta x. \tag{B.5}$$

APPENDIX C

Calculations for the Maximum Velocity of the Load on the Linear Positioner
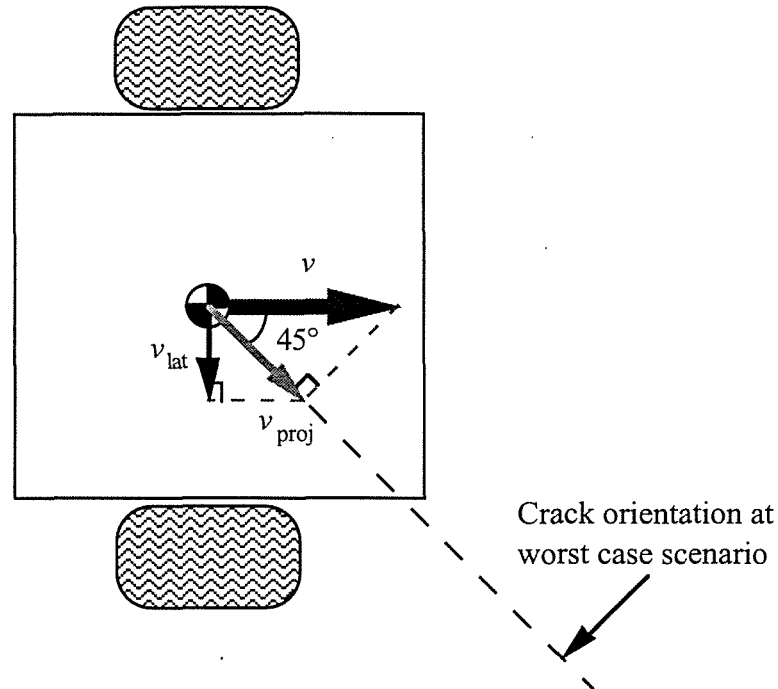
Figure C.1 *Velocity Diagram of the TMRR*

Assuming the worst case scenario, a speed of 3 km/hr (2 mph) and a crack orientation of 45° are used for the calculations. As the TMRR moves forward at a speed of 3 km/hr (2 mph), the projection of this velocity vector along the crack, which is oriented at 45° is given by

$$v_{proj} = v\cos 45° \tag{C.1}$$

where v is the forward velocity of 3 km/hr (2 mph). Thus the projected velocity along the crack is 2.3 km/hr (1.4 mph). The speed of the carriage on the linear positioner that is necessary to track the crack is the lateral component of the projected velocity, and the

lateral velocity is

$$v_{\text{lat}} = v_{\text{proj}} \cos 45°$$

(C.2)

which yields a transverse speed of 1.6 km/hr (1 mph). This is the maximum linear speed

of the carriage on the linear positioner required for our application.

APPENDIX D

Manufacturer's Specifications for the Linear Positioning System

Drive Screws and Nuts

General Description
The screws used in Jasta actuators and positioners are "rolled" screws with a lead accuracy of ±.003 to ±.005 inch per foot of screw. "Ground" screws are available at a considerably higher price. The ground screw lead error is ± 0.0005 inch per foot of length, non-accumulative.

Lubrication
All Jasta actuators and linear positioners are factory lubricated with the proper screw lubricants for the life of the particular unit. For special applications; i.e., extremely high or low temperature or for vacuum operation, Jasta will lubricate accordingly.

Materials
Jasta Vee and Acme screws are cold-rolled 1018 steel or stainless steel if required. Most Jasta Ball screws are 1018 steel hardened to RC65.

Designation
The screw designation is used throughout this catalog. To interpret the designation, the letter (V, A, or B) designates the screw type, and the number following the letter designates the screw pitch, as described in the table below.

Pitch
This designates the number of threads per inch of screw (i.e., V20 is a Vee thread form with 20 threads per linear inch).

Lead
The screw lead is the linear distance the screw drive nut will move for each revolution of the screw (i.e., V20 = 1.00 inch/20 threads = 0.05 inch/revolution.

Efficiency (e)
The screw efficiency is determined by measuring the torque required to rotate the screw in order for the drive nut to lift a known load.

$$e = \text{Load (Lbs.)} \times \text{screw lead} \times 16 \text{ oz/lb.}$$
$$\div 2\pi \times \text{Torque (oz/in.)}$$

Specifications
Jasta linear actuators and positioners utilize the drive screws listed in the Table below. Each actuator is given a catalog size reference . For example:

Size 1 = Mite, Mini-Pulse, PP I, and Mini-Jac
Size 2 = Act I, Act II, Jasta-Jac, PPII and PPIII
Size 3 = Brute

Type
Jasta actuators and positioners are available with three types of screws: Vee thread, Acme thread, and Ball thread. Vee thread screws offer very fine leads and are quite useful for applications not requiring high velocities, high loads or high duty cycles. Acme screws are very useful in applications where noise is a consideration because they are generally quieter running. The acme also offers a wide variety of screw diameters and leads, can be self-locking and is less expensive when compared to ball screws. Ball screws have many advantages over acme and vee thread screws such as: higher efficiencies, higher speeds, higher loads, higher duty cycles, less friction and longer life.

*Efficiencies in the table below are based on Delrin AF drive nuts on the Vee and Acme screws, and ball nuts on the ball screws.
For Bronze nuts reduce the efficiency of Delrin AF nuts in the Table by 5%..

Drive Screw Data

Designation	V40	V20	A10	A5	A2	A1	B8.5	B8	B5	B2
Thread Form	Vee	Vee	Acme	Acme	Acme	Acme	Ball	Ball	Ball	Ball
Pitch	40	20	10	5	2	1	8.5	8	5	2
Lead	.025"	.05"	.10	.2"	.5"	1.00"	.118"	.125"	.20"	.50"
Efficiency	.18	.25	.30	.40	.60	.65	.90	.90	.90	.90
Backdrive*	No	No	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Max. Duty	40%	50%	60%	60%	70%	80%	100%	100%	100%	100%
Actuator Size 1 Diameters	.375"	.375"	.375"	.375"	.375"	.375"	.397"	.375"	N/A	N/A
Actuator Size 2 Diameters	N/A	.50"	.50"	.50"	.50"	.50"	N/A	N/A	.625"	.50"
Actuator Size 3 Diameters	N/A	.75"	.75"	N/A	N/A	N/A	N/A	N/A	.75"	.75"

*The higher the lead of the screw the less effort required to backdrive either the screw or the nut. As a rule, the lead of the screw should be more than 1/3 the diameter of the screw to satisfactorily backdrive.

N/A = Non-Applicable

4

Drive Screws and Nuts

Screw Linear Velocity

All screw types are limited as to the speed to which a screw may be rotated. At a particular speed for a particular screw, the 'critical velocity' of the screw will be reached, causing the screw to whip and vibrate, thereby becoming unstable and unusable. This critical velocity is a result of screw rotational speed, screw diameter, and screw length (refer to page 11).

Screw Life

The operating life of Jasta linear actuators and positioners is dependent on numerous factors: stroke velocity, thrust, load, moment, environment, screw type, motor type and duty. Upon selecting an actuator or positioner for a continuous duty operation, choose a ball screw and nut along with a motor rated for continuous duty (100%). Acme screws (with leads below .5") and nuts are relatively high-friction with efficiencies of less than 60%, and generate heat at high linear velocities and high duty cycles. Ball screw/nut combinations have efficiencies of 90% and are designed for higher loads and speeds. Once an actuator is properly selected, the life can range to a high-end of 50 million cycles for a ball screw model under normal conditions.

Drive Nuts
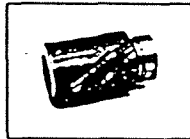
Ball Nut

Delrin AF Nut

Anti-Backlash
Acme Nut

Bronze Nut

Ball Nut

Jasta "Ball" nuts utilize high quality steel ball nuts with recirculating steel balls for long life and heavy loads. Normal backlash is approximately .002" to .004".

Delrin AF Nut

The "Delrin AF" drive nut is a combination delrin and teflon plastic drive nut, utilized in all Jasta acme and vee threads for actuator loads under 500 lbs. Some advantages of this nut are its quiet running and low friction operation.

Anti-Backlash Acme Nut

The Delrin AF "ZN" nut (Anti-backlash Acme Nut) is available with a zero backlash design. This is accomplished by preloading the nut by spring compression.

Bronze Nut

A Bronze drive nut is used on acme screws in applications where thrust requirements exceed 500 lbs. It's main advantage is its heavy load capability.

Gearing

Jasta actuators are driven by three types of gear reduction:

1. Synchronous (timing) belts and pulleys
2. Worm gear drives
3. Inline drive (no gear reduction)

Designation

All gearing has a specific designation thru-out this catalog. "T" represents a "timing belt drive". "W" represents a "worm gear" or "right angle" drive, and "D" represents a "direct drive."

Ratios

Ratios are described by the number following the letter representing the type of gearing.
For instance: T2 = Timing belt (2:1 ratio)   D1 = Direct drive (1:1 ratio)   W10 = Worm gear (10:1 ratio)
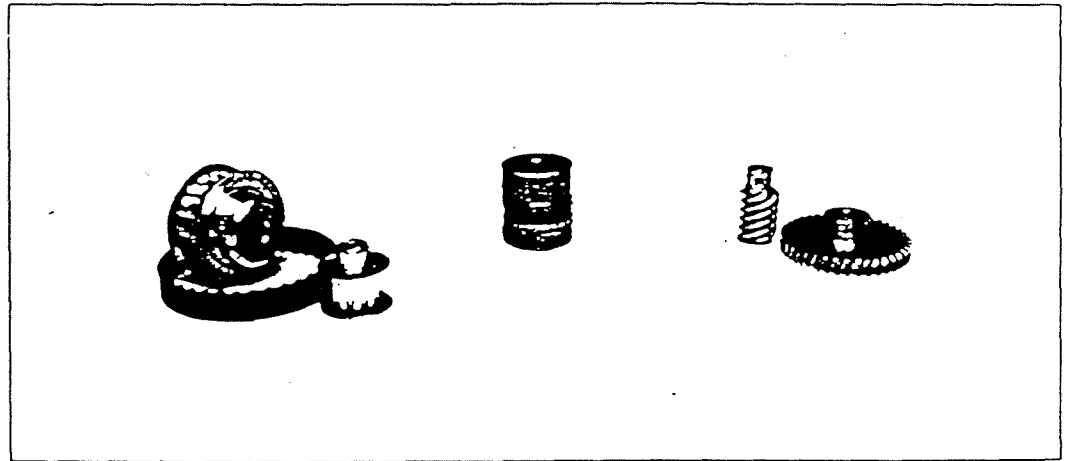
Synchronous Belts and Pulleys:	Worm Gear (Right angle) Drives:	Inline Drive (Direct Coupled):
The majority of Jasta actuators use a no-slip, very low backlash, synchronous belt drive known as timing belts. Timing belts and pulleys have the following advantages over spur and worm gears: A) Long life B) Low backlash C) No lubication required D) Quiet running E) Unaffected by minor mis-alignments F) High efficiency – 90% +	Jasta utilizes worm gear drives in the Jasta Jac and Mini Jac series of actuators. The advantage of the worm gear is a high ratio in a small space and high loading capability; however, the gear efficiency is low ~50%. Jasta worm gear ratios available: • 10:1 • 20:1 • 40:1 *The 40:1 ratio with a ball screw will not backdrive under load.	Jasta Pulse Power series use a direct drive with no gear reduction. The motor shaft is coupled directly to the drive screw with a high quality zero-backlash coupling. The advantages with direct inline coupling are: A) Zero-backlash B) Long life C) No gear noise D) Highest efficiency

Motor Data

The motors utilized by Jasta on linear actuators and positioners fall into one of four categories:

1. Permanent magnet, brush type direct current motors, which are available in 12, 24, and 90 VDC.

2. Permanent magnet, brush type servo motors; which are available in 24 and 90 VDC.

3. Permanent magnet, direct current two and four phase step motors, with 200 or 400 steps per revolution.

4. Brushless servo motors with complete control systems, which are available in 115 VAC or 230 VAC input to amplifier.
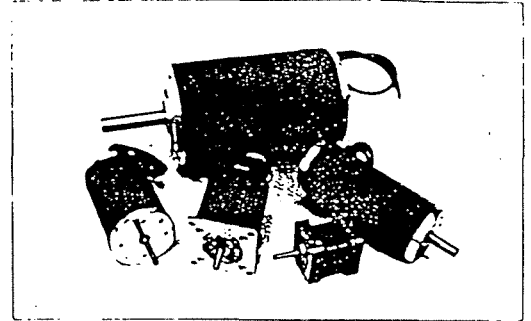
*Jasta will also install the motor of your choice if it meets Jasta's installation requirements.

Motor Type	Advantages	Disadvantages
Permanent Magnet DC Brush	•Economical •Fast acceleration •High speed •High Torque in a small frame •Maximum Torque at stall •No starting capacitors required •Speed controllable	•Brush wear •Electromagnetic Interference
Permanent Magnet DC Servo	•All of the above •Better precision •Better brushes	•Same as above
Step Motor	•Long life •No brushes •Moderate cost •High Torque •Continuous duty •Precision	•Low to moderate speed •Electromagnetic Interference •Hot running
Brushless Servo	•No brushes •High Speed •High Torque in a small frame •Fast acceleration/deceleration •Incremental or Absolute encoder •Speed and/or positioning control •Continuous duty •Long life	•Most expensive

7

U.S. DIGITAL

E2 Series
Technical Data, Rev. 1.28, Feb. 1995

Quick Assembly
Optical
Encoder

The E2 optical incremental shaft encoder is a noncontacting rotary to digital position feedback device designed to easily mount to an existing shaft.

The internal monolithic electronic module converts the realtime shaft angle, speed and direction into TTL-compatible outputs. Simplicity and low cost make the E2 ideal for both high and low volume motion control applications.

The E2 consists of four parts: base, cover, code wheel and encoder module. The encoder module incorporates a lensed LED light source and monolithic photodetector array with signal shaping electronics to produce the two channel bounceless TTL outputs.

The hub diameter is specified when ordering to adapt to any shaft diameter up to 3/8". Standard diameters are stocked. Quick turn around time is also offered for any special order diameter.

The cover is available in three configurations: The standard is a solid flush back which can accommodate a shaft length up to .57". Option 'H' specifies a .375" diameter hole in the flush back for the shaft to pass through. This hole diameter is .500" when the 3/8" diameter hub is specified. Option 'E' provides a cylindrical extension making room for a 3/4" shaft length.

The base provides mounting holes for 2 (2-56 or 4-40) screws in a .750" bolt circle or 3 (0-80) screws in a .823" bolt circle. Option-3 makes all five of these hole diameters .125". If desired, the two .096" diameter recesses will mate with matching aligning pins. The .438" diameter center hole can also mate with a motor boss. The standard base is flat. Option 'A' adds a .497" diameter alignment shoulder designed to slip into a .500" diameter recess centered around the shaft.

Features
- Quick and simple assembly & dis-assembly
- Rugged screw-together housing
- Low cost
- Accepts ±.010" axial shaft play
- Small size
- Tracks from 0 to 100,000 cycles/sec
- 96 to 1024 cycles/rev.
- 384 to 4096 codes per revolution
- 2 channel quadrature TTL squarewave outputs
- Optional index (3rd channel)
- -40 to +100°C operating temperature
- Compatible with HP HEDS-5500
- Fits shaft diameters to 2mm to 3/8"
- Single +5v supply
- Flush back, through shaft hole, or extended back
- Flat or self-aligning base
- Also adapts to 1.812" bolt circle (2 or 3 holes)

Electrical Specifications:
See our HEDS Optical Encoder Module data sheet

Bottom View   End View   Side View   Top View
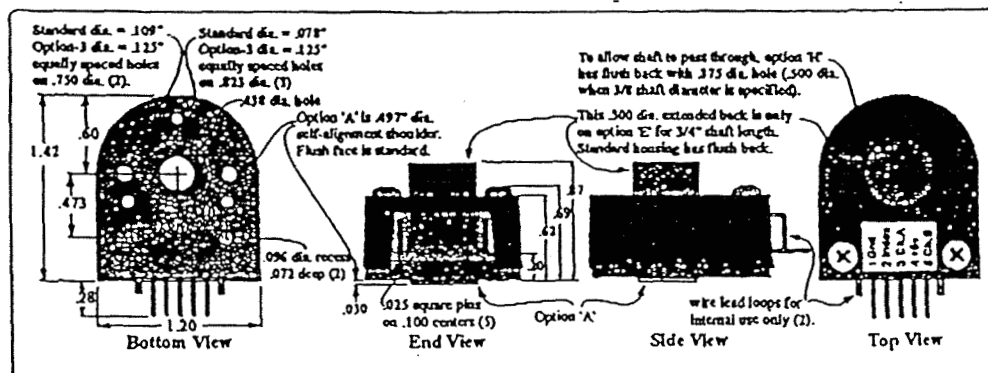
Mechanical Specifications

Parameter	Dimension	Units
Moment of Inertia	8.0 x 10⁻⁴	oz-in-s²
Hub Set Screw Size	3-48	inches
Hex Wrench Size	.050	inches
Encoder Base Plate Thickness	.135	inches
3 Mounting Screw Size	0-80	inches
2 Mounting Screw Size	2-56 or 4-40	inches

Mechanical Specifications

Parameter	Dimension	Units
3 Screw Bolt Circle Diameter	.823 ±.005	inches
2 Screw Bolt Circle Diameter	.750 ±.005	inches
Required Shaft Length	.445 to .570*	inches
with 'E' Option	.445 to .800*	inches
with 'H' Option	>.445*	inches

* Add .125" to the required shaft length when using the P-option or R-option adapter plates.

Phone (360) 696-2468   Sales (800) 736-0194   Fax (360) 696-2469
For Product Information call our automated Fax Service at (360) 696-3836 from your fax machine.
U.S. Digital Corp. • 3800 N.E. 68th St., Suite A3 • Vancouver, WA 98661-1353

Selecting an Actuator

Curves calculated for smallest root (minor) diameter of screws normally used by JASTA, and include a 20% factor of safety.

Jasta Screw Diameters
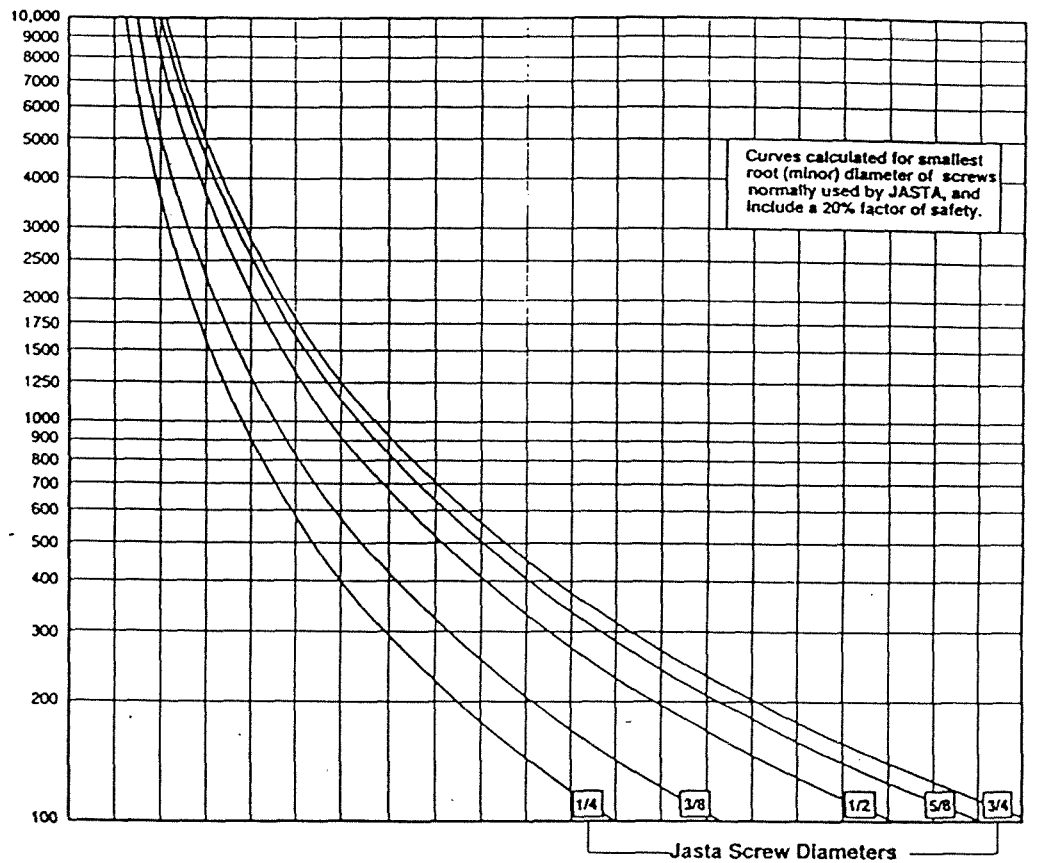
Actuator	0	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68	72	76	80	84
LP	0	10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200	210

STROKE (inches)

11

Selecting an Actuator

Step 9: Using the motor curve selected from STEP 8, verify that the required torque and speed fall into the correct duty cycle area, as indicated within the curve area by being either shaded or nonshaded; i.e., 100%, 60%, 30%, or 10%.

Step 10: Use the table in FIGURE 7 to determine the resolution and repeatability of the actuator selected.

Screw	Screw Lead (in.)	Repeatability ± in.			Resolution (in.)
		*24VDC Motor	*90VDC Motor	Step Motor	200 steps per screw rotation
V40	.025	.003	.005	.0005	125 X 10⁻⁶
V20	.05	.003	.005	.0005	25 X 10⁻⁵
A10	.10	.003	.006	.0005	5 X 10⁻⁴
B8-B8.5	.125/.118	.003	.006	.0005	6.25 X 10⁻⁴
A5-B5	.20	.004	.006	.0005	1 X 10⁻³
A2-B2	.50	.008	.010	.0005	2.5 X 10⁻³
A1-B1	1.00	.010	.015	.0005	5 X 10⁻³

*Repeatability figures are based on positioning by limit switches.

Figure 7

Step 11: Consider the environmental requirements that may be involved in the particular application; i.e., dust, water, wash downs, extremely high or low temperature, corrosives, etc.
Josta actuators can handle many extreme conditions. Check with the factory for special requirements.

Step 12:
Ensure the load is acting down the centerline of the translating tube as shown in FIGURE 8. Only the linear positioners can take bending moments. External linear guides should be incorporated to eliminate bending moments on all other Josta actuators.
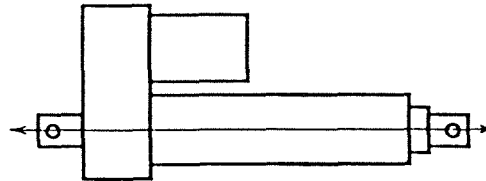
Figure 8

Step 13: Mounting requirements can be determined by referring to the "Mounting Components" section on pages 37-43.

Step 14: Refer to pages 54-56 for options that may be added to enhance actuator performance.
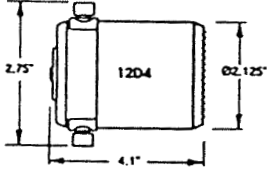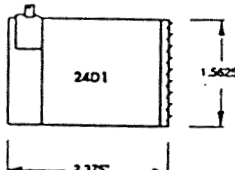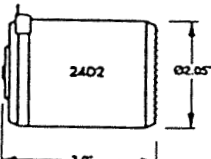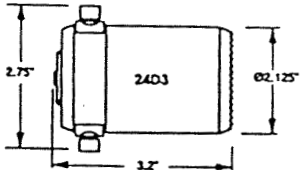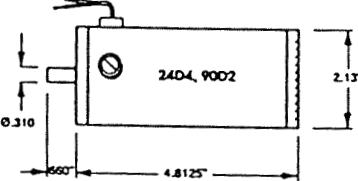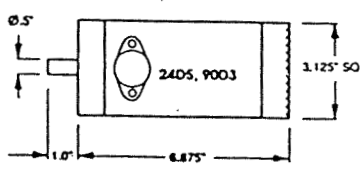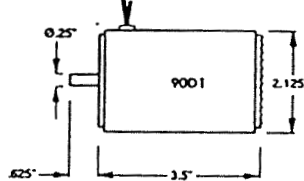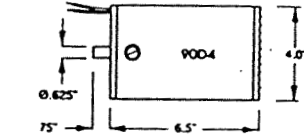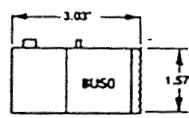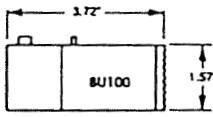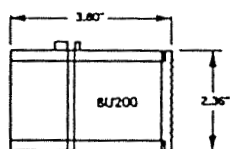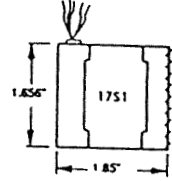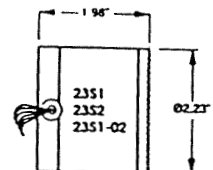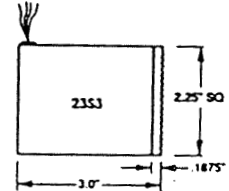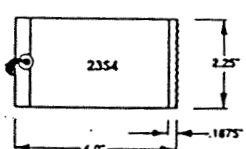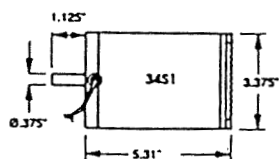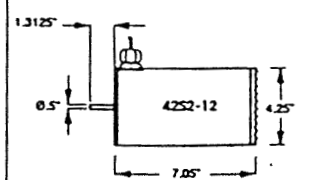
Motor Specifications

| DC Motors | | | | | | | | | No |
Motor Number	Voltage (DC)	Stall Torque (in.-oz)	Cont. Torque (in.-oz)	Stall Current (Amp)	KT (in.-oz /Amp)	KE (Volts/ Krpm)	Resist-ance (Ohm)	Induct-ance (Mh)	Load Speed (rpm)
12D4	12	204	32	43.7	4.33	3.21	.27	.4	3700
24D1	24	24	4	3.3	4.0	3.2	4.1	3.3	7300
24D2	24	42	7	8	5.3	3.7	3.1	5.1	5100
24D3	24	107	16	13.9	7.8	5.77	1.73	2.54	4087
24D4	24	275	40	33.3	8.4	6.2	.72	4.0	4300
24D5	24	1000	90	136	7.35	5.56	.5	5.0	3500
90D1	90	90	5	2.0	46.0	34.0	45.0	4.0	2500
90D2	90	200	40	4.5	44.0	33.3	18.5	18.0	2700
90D3	90	650	81	13	59.1	45.0	6.8	12.0	2000
90D4	90	3800	416	50	61.0	45.1	1.4	6.5	2000

| Step Motors | | | | | | | | | No |
Motor Number	Voltage (DC)	Stall Torque (in.-oz)	Cont. Torque (in.-oz)	Max. Allow. Amp/Ph.	Min. Req'd Volts/Ph.	KE (Volts /Krp)	Resist-ance. (Ohm)	Induct-ance (MHz)	Load Speed (rpm)
17S1	12-60	44.4	•	1.2	4.0	NA	3.3	3	•
23S1	12-60	53	•	1	5.0	NA	5.1	10	•
23S1-02*	160	60	•	1.0	5.0	NA	5.0	8	•
23S2	12-60	53	•	3.8	1.2	NA	.33	.6	•
23S3	12-60	100	•	4.7	1.6	NA	.35	.8	•
23S3-06*	160	150	•	2.9	3.4	NA	1.2	2.9	•
23S4	12-60	150	•	4.6	2.2	NA	.4	1.1	•
34S1	12-60	450	•	4.8	3.3	NA	.65	4.2	•
34S3-11*	160	450	•	5.5	2.9	NA	.52	2.2	•
42S2-12*	160	1100	•	6.1	3.6	NA	.6	3.6	•

Note: Step Motor performance for speed and torque is based on the type of control used. See motor curves on pages 13-14. *These motors come equipped with a six foot length of cable and are encoder ready.

| Brushless Servo Motors | | | | | No |
Motor Number	Voltage (AC)	Peak Torque (in./oz.)	Cont. Torque (in.-oz)	Peak Current (Amps)	Load Speed (RPM)
BU50	110/220	68	23	2.9	4500
BU100	110/220	135	45	7.1	4500
BU200	110/220	270	90	8.4	4500

Note:

Brushless Servo Motor Specifications are based in conjunction with the use of the JBU100 Control as found on pages 63-64.

13

Motor Dimensions

DC Brush Motors	DC Brush Motors	Step Motors

Note: ⸾ represents the side of motor mounted to actuator.

DC Step Motor Torque vs. Speed Curves
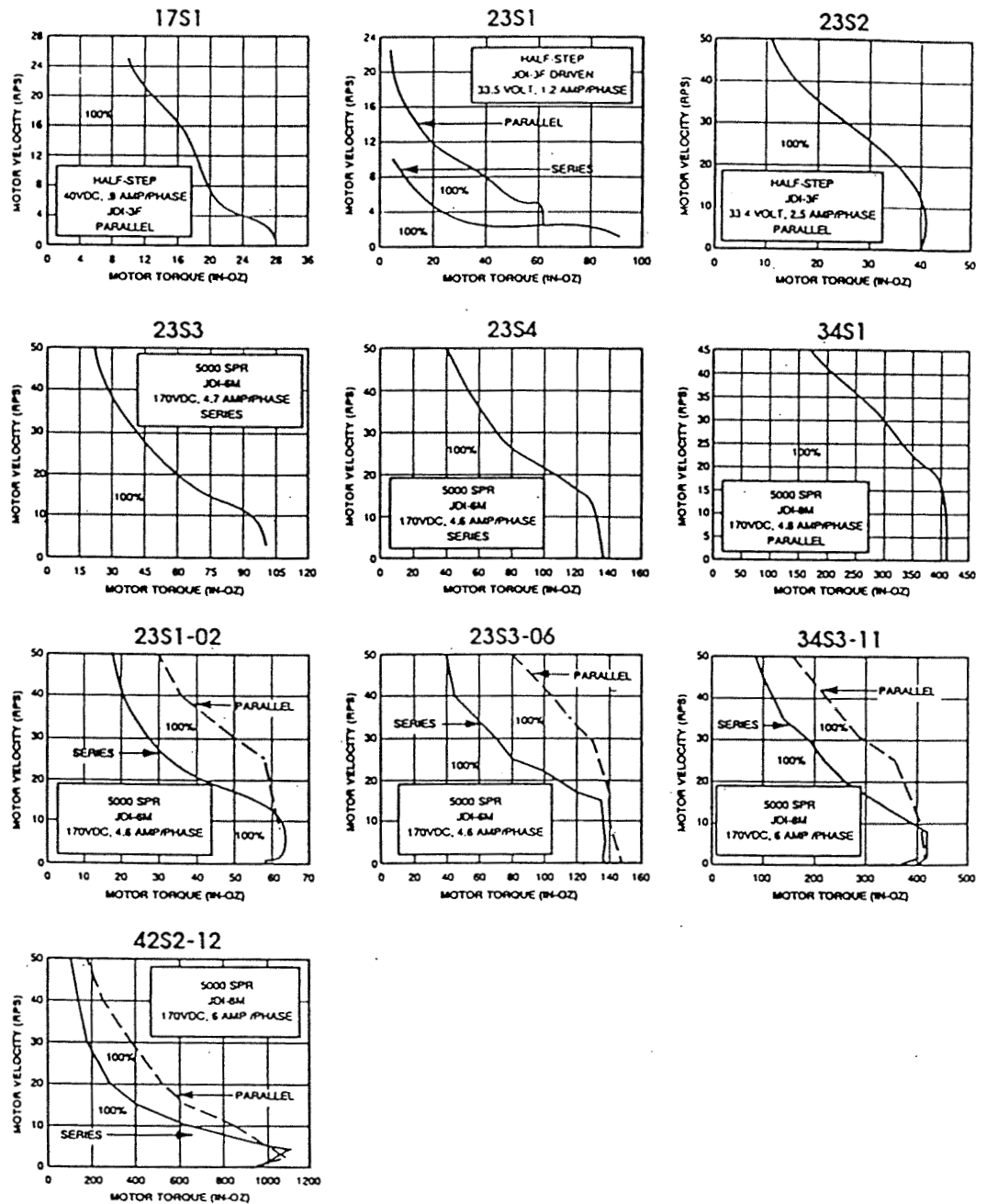
Note: Percentages under curve represent motor duty cycle (assuming proper drive and cooling systems are used).

Linear Positioner Series Introduction

General
Josta-Dynact's Linear Positioner or "Rodless Cylinder" series are high quality units designed for applications where long stroke lengths, high velocities, and precise resolution and repeatability are necessary.

Stroke length
See the table below for all stroke lengths available per series. Final stroke length depends on screw velocity desired.

Gearing
Josta-Dynact's standard gear ratios for the linear positioners are also listed in the table below. The letter "D" designates "direct drive" and The letter "T" represents a "timing" belt drive. All numbers following each specified letter represent the ratio available. Please refer to page 6 for more information.

Drive screws and Drive Chains
"Screw" drive linear positioners use 3 types of drive screws: Vee (V), Acme (A), and Ball (B). The number after the letter designates the screw pitch, as listed in the table below. Please refer to pages 4-5 for more information on drive screws. "High-velocity" or "Cable chain" drive linear positioners use drive chains for maximum velocity. (see page 47).

Motors
Josta-Dynact offers a wide variety of motors. Any number preceding the letter "D" relates to the voltage of the motor, and the number preceding the letter "S" relates to the frame size of the step motor. The number following the letters "D", "BU", and "S" designates the size of the motor relative to the sizes within its category. Please refer to pages 13-16 for more information.

Model	*Stroke Length (Approximate)	Gearing	Drive Screws	Motors
LP 050	0-20 Inches	D1	V40, V20, A4, A2, A1	17S1, 24D1 & BUS0
LP 100	0-72 Inches	D1, T1, T2 & T2.67	V20, B8, B8.5, A10, A5, A2 & A1	24D4, 9DD2, BU100, & 23S4
LP 150	0-72 Inches	D1, T1 & T2	V20, A10, A5, A2, B5 & B2	24D5, 9DD3, BU200, & 34S1
LP 100 HV	0-20 Feet	D1 & T2	N/A	24D4, 9DD2, BU100, & 23S4
LP 150 HV	0-20 Feet	D1 & T2	N/A	24D5, 9DD3, BU200, & 34S1

Options
Various options adding to the operation of the linear positioners are (BR) Electromechanical Fail-Safe Brake, (MSR) Magnetic Switch Relay, (MS) Magnetic Reed Switch, (ES) Environmentally Sealed, (ZN) Anti-Backlash Nut, (LZ) Low-Backlash Ball Nut, (EN) Optical Encoder, (CN) Brake or Encoder Canister, (CND) Brake and Encoder Canister, and (GRS) Guide-Rail Supports. Please refer to pages 53-55 for more information.

Mounting
"Screw" drive linear positioners can also be manufactured where the motor is mounted parallel to the drive screw, as in the case of an actuator for instance. This alternative method of operation is available for the linear positioner "screw" drive series only. (refer to page 51). "High-velocity" linear positioners are mounted where the motor is perpendicular to the cylinder body.

Clamp-On Foot Mount (CFM) option: The "Clamp-on Foot Mount" (CFM) may be substituted for the standard linear positioner foot mounts. (See pages 38, 41 and 42).

44

Linear Positioner Features

Carriage table for mounting. Provides superior load carrying performance.

Shock absorber cushions the carriage when it comes to the end position.

Polyurethane seal for durability and long life.

Screw Drive

Vee, acme, or ball drive-nut is guided by a key which carries the positioner's load and moments.

Vee, acme, or ball thread form screw (lifetime lubricated).

Cable-Chain is zero-backlash and steel reinforced

Mounting foot

Adjustable idler for cable chain tension.

Linear Positioner Features

Zero-backlash
coupling

Dual bearings

- Aluminum extrusion for durability,
non-corrosiveness, and long-life.
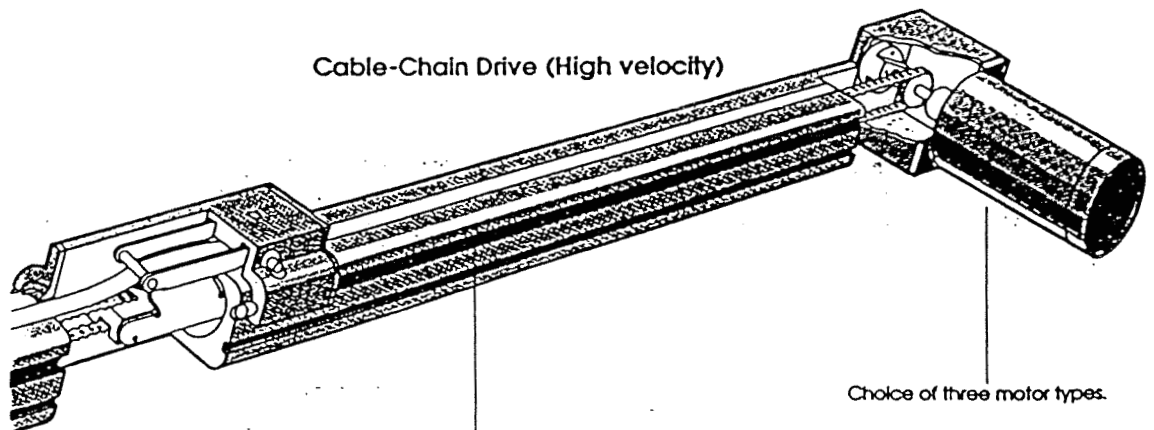
Cable-Chain Drive (High velocity)

Choice of three motor types.

Switch track for mounting mechanical or
magnetic switches.

Linear Positioner Specifications

Screw Drive
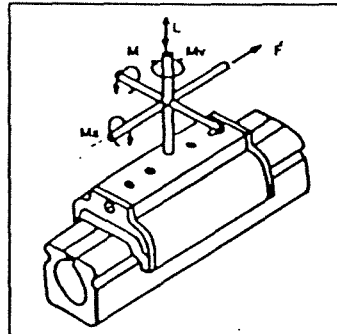
Load, Thrust, and Bending
Moments Diagram

Load, Thrust, and Bending Moments Specifications

| Model Number | Maximum Bending Moment | | | Maximum Load/lbs. (L) | Maximum Thrust/lbs. (F) |
	Pitch (M) in./lbs.	Roll (Mx) in./lbs.	Yaw (My) in./lbs.		
LP 050	10	5.5	.6	10	10
LP 100	100	55	30	60	200
LP 150	500	175	200	180	400

Weight Specification Table

Model Number	Base Weight w/Motor	Weight per Inch of Stroke
LP 050	1.3 lbs.	.04 lbs.
LP 100	6.2 lbs.	.23 lbs.
LP 150	19 lbs.	.50 lbs.

Belt Drive (High-Velocity)
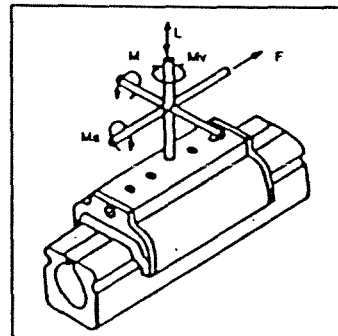
Load, Thrust, and Bending
Moments Diagram

Load, Thrust, and Bending Moments Specifications

| Model Number | Maximum Bending Moment | | | Maximum Load/lbs. (L) | Maximum Thrust/lbs. (F) |
	Pitch (M) in./lbs.	Roll (Mx) in./lbs.	Yaw (My) in./lbs.		
LP 100 HV	100	55	30	60	50
LP 150 HV	500	175	200	180	60

Cable Chain Specification Table

Model Number	Drive Pulley Circumference	Maximum Thrust
LP 100 HV	2.50 Inches	100 Lbs.
LP 150 HV	4.00 Inches	100 Lbs.

Cable Chain

Polyurethane

Stainless
Steel Cable

Sprocket
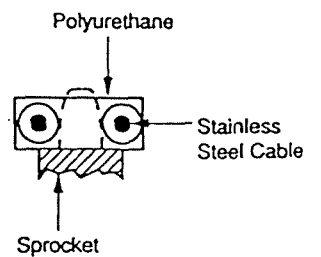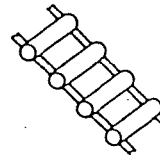
Weight Specification Table

Model Number	Base Weight w/Motor	Weight per Inch of Stroke
LP 100 HV	6 lbs.	.18 lbs.
LP 150 HV	18 lbs.	.34 lbs.

dasa

47

Linear Positioner Dimensions (Screw Drive)

Parallel Mount configuration

VIEW A
TOP

VIEW B
SIDE

VIEW C
MOUNTING TABLE

LP100 DIMENSIONS

A	B	C	D	E	F	G	J
2.5	1.5	.5	.25	.25W & .25L	1.5	1.5	.50
L			P	Q	S	T		V
1/4-20 UNF X .25 DP, 4 PLCS	1/4-20 UNC X .25 DP, 2 PLCS	.50	.05	1.25	1.25		1.25 + §
W	X	Y	Z	AA	BB	CC	DD
2.875	2.5	1.5	.5	4.75	1.5	.	.
EE	FF	GG	HH	II	JJ	KK	LL
1.5	6.05 + §	6.75 + §	1.25	2.5	.5	.05	.125
MM	NN	OO	PP	QQ	RR	SS	TT
1.75	1.50	2.125	2.375	1.5	1.625	.50	.75

*For motor dimensions, refer to page 14   § = Stroke

LP150 DIMENSIONS

A	B	C	D	E	F	G	J
2.4	1.5	.5	1.25	.25W & .25L	2.25	2.25	1.25
L			P	Q	S	T		V
5/16-18 UNC X .375 DP, 2 PLCS	1/4-20 UNC X .25 DP, 4 PLCS	1.5	.25	1.50	1.50		12.625 + §
W	X	Y	Z	AA	BB	CC	DD
4.5	2.65	2.25	.5	1.50	2.00	.	.
EE	FF	GG	HH	II	JJ	KK	LL
1.5	8.05 + §	14.575 + §	1.25	2.5	1.40	.125	1.25
MM	NN	OO	PP	QQ	RR	SS	TT
2.25	1.95	2.625	2.75	2.125	1.50	.75	1.15

Actuator and Linear Positioner Series Options

General

Jasta offers the following as options for the actuators and positioners. These items enhance overall performance, protect operating components, and increase longevity.

Overload Protection

Jasta actuators and positioners should not be continually operated against the end stops, thereby putting the motor in "stall". Overload protection devices such as mechanical limit switches, magnetic reed switches, and linear potentiometers are some of the options which should be incorporated into the actuator or positioner to prevent overload.

Description:

BR - Electromechanical Brake: Required on all units with a ball screw nut combination or an acme lead of .5 inch or over in applications where load is acting opposite to drive force, tending to "backdrive" the system when stopped.

ES - Environmentally Sealed: A special sealing on the actuator or positioner to guard against very tough environmental conditions; i.e., snow, heavy rain, washdowns, sandstorm, etc.

MSR- Magnetic Switch Relay: The MSR is an add on package which converts the magnetic reed switch signal into a stand alone limit switch package. It can be wired directly in line with the motor and is capable of switching up to 15 amps - enough to handle full motor current on most of Jasta's actuators.

MS - Magnetic Reed Switches: Reed switches can be adjusted at any position along the actuator or positioner's switch track. A magnet within the drive nut of the actuator "trips" the switch as it passes. A magnet on the LP carriage "trips" the switch as it passes. Rated: 0.15 amp maximum at 100 VDC. Rated life: 5 million cycles. Refer to page 55 for more information regarding dimensions.

ZN - Antibacklash Acme Nut: Removes backlash from the plastic drive nut and is used in precise positioning applications where lost motion cannot be tolerated. Please refer to page 5 for more info.

LZ - Low Backlash Ballnut: The ballnut is individually fitted to the screw, giving maximum backlash of 0.002" or less.

GSR- Guide Rail Supports: Used only for the Linear Positioner series LP100 and LP150. These support rails serve in accomodating higher loads (See page 48).

EN - Optical Encoder: An encoder is fitted to the actuator or positioner for servo or step motor closed-loop systems. Refer to page 55 for more information regarding dimensions.

*LPO - Linear Potentiometer: Position of the actuator stroke can be determined by resistance output proportional to the actuator stroke position. The LPO is linear to ± 1% of total stroke. LPO output is 1500 ohms per inch of stroke length. Stroke availibility of 2,3,6,9,and 12.

*LPO is not available with Jasta's "Linear Positioners".

CN- Encoder or Brake Canister: A protective canister which houses an encoder or brake option and conveniently includes quick connect/disconnect terminals for all wiring associated with the encoder or brake option. Refer to page 55 for more information regarding dimensions.

CND- Encoder and Brake Canister: Same function as the "CN" option but designed to house both the encoder and brake options. Refer to page 55 for more information regarding dimensions.
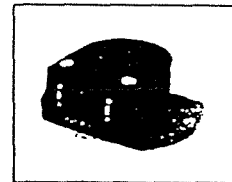
Optical Encoder
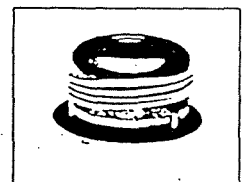
Electromechanical Brake
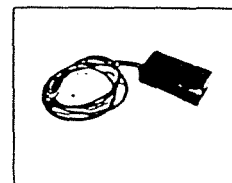
Magnetic Switch Relay
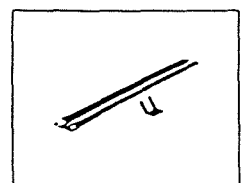
Linear Potentiometer
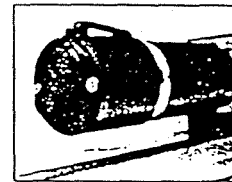
Brake or Encoder Canister

Magnetic Reed Switch

54
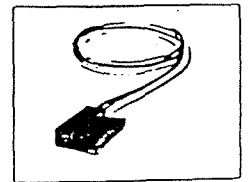
Actuator and Linear Positioner Option Dimensions

Motor, Brake and Encoder Canister Dimensions (CND)

Encoder/LPO Connector (9 Pin)

Motor & Brake Connector (8 Pin)

"2.375"

.4375"

2.0625"   2.5"
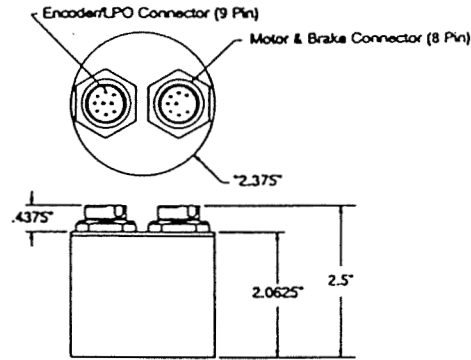
"This dimension changes to 3.00" for a size 2

Motor, Brake or Encoder Canister Dimensions (CN)

Encoder/LPO or Brake Connector

"2.375"

.4375"

1.3125"   1.75"

"This dimension changes to 3.00" for a size 2

Electromechanical Fail-safe Brake (BR)

Ø.172", 4 PLCS.
2.844" DBC

Ø.7812"   .960"

Ø2.260"   Ø3.235"

Size 1

Ø.200", 4 PLCS
3.125" DBC

Ø.362

Ø2.810   Ø3.5"

Size 2   1.905"

Optical Encoder (EN)

2.0625"

Mounting Base
Encoder   .625"

.125

1.375"

.500"

Wire Leads

1.125"   Wire Connector

Encoder Compliments are available

Magnetic Switch Relay (MSR)

4.0"

2.0"

1.37"   WIRES (12.0" ±.250")
"Wires are in no specific order

Magnetic Reed Switch (MS)

.906"   Ø.156 TYP.
.551"
.115"
.250"   .551"
.232"

WIRES (12.0" ±.250")
"Wires are in no specific order

Note: For more information regarding option specifications contact Jasta.

55

Option Locations

Magnetic Reed Switches (MS)
are adjustable along switch track

Brake (BR)

Magnetic Switch Relay (MSR)

Brake and/or Encoder Canister (CN or CND)

Quick Connect Terminals

Optical Encoder (CN)

All of the option mounting locations as shown in these configurations for the linear actuator are the same for the linear positioner series.

"JDI" Series Step Motor Controls

- Includes a powerful built-in indexer, driver, and power supply
- Optically coupled step and direction inputs for reliable operation
- Easy to learn English-like instructions
- Built-in editor allows creation of up to 88 different motion programs
- RS-232 communications allows daisy chaining of multiple devices
- Home, extend & retract limits, and multiple programmable inputs and outputs
- Encoder feedback capabilities (JDI-6M & 8M only)
- Short circuit and over temperature protected
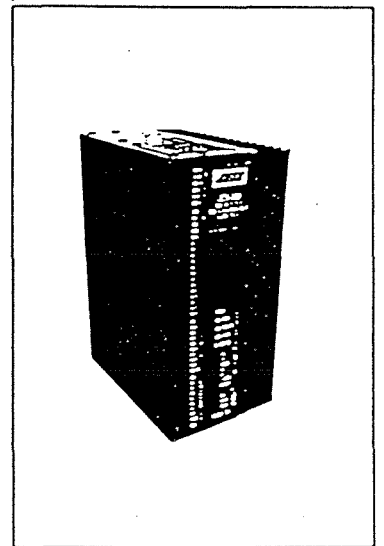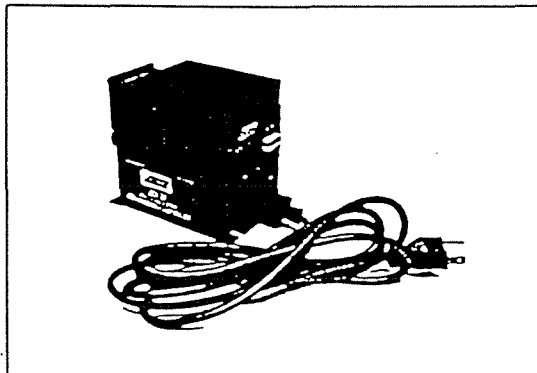
General

The "JDI" series step motor controls are fully compatible with all Jasta-Dynact actuator series, utilizing the step motor as the driving motor. These units are of the highest quality and finest workmanship and are designed to deliver optimum performance to all Jasta-Dynact actuators.

61

"JDI" Series Specifications

Common Specifications	JDI-3F	JDI-6M & JDI-8M
Drive Type	2 phase, bipolar, constant current, MOSFET chopper, 20 KHz fixed.	
Resolutions	200, 400.	200, 400, 1000, 2000, 5000, 10000, 18000, 20000, 21600, 25000, 25400, 25600, 36000 50000, 50800.
Power (Inputs)	115-230V, 50/60Hz, 50VA.	90-135VAC, 50/60Hz, +5VDC logic.
Power (Outputs)	0.2-2.8 amp/phase programmable.	JDI-6M=0.2 to 6 amps JDI-8M=2 to 8 amps
Protection (Short circuit)	Phase to phase, phase to ground.	
Protection (Over Temperature)	Internal air temperature exceeds 140° F (60° C).	
Automatic Current Limit	50% of running current when motor is stationary.	
Fault Output	Sinking output to OUTCOM, 5-24VDC, 60mA maximum. Disable LED on.	
Low Power Mode (Auto reduce)	Programmable "Hold" current 1% resolution.	DIP switch enabled. Current drops to 50% of selected value if no step pulses are received in one second.
Environmental (Temperature)	Drive max. 130° F (55° C), storage -50° F to 185° F (-45° to 85° C).	Drive heatsink max. 140° F (60° C); Motor case max. 212° F (100° C); Storage -40° F to 185° F (-40° C to 85° C)
Environmental (Humidity)	0-95% non-condensing.	
Additional Specifications		
Operational		
Acceleration	Programmable ramps.	Optimal non-linear mathematical function or programmable.
Step Accuracy	±0 steps.	
Position Range	±17 million steps.	±2.1 billion steps.
Speed Range	0 to 23,000 pulses/second.	0 to 750,000 pulses/sec. ±1% max. speed.
Communications		
Type	RS-232 serial or RS-422.	RS-232C serial, 3 wire implementation (Tx, Rx, Ground).
Baud Rates	300, 1200, 2400, 4800, 9600, 19200.	1200, 2400, 4800, 9600, 19200.
Mode	Full duplex.	
Format	8 data bits; 1 stop bit, no parity; ASCII characters.	
Multi-Axis	Daisy chain 32 indexers on RS-422.	Daisy chain up to 36 indexers from a single host RS-232C port.
Inputs		
Type	Optically isolated, 5VDC. CW, CCW, HOME.	Optically isolated, TTL or 5-15VDC
Limits		
Programmable	Five.	Thirteen.
Interrupts	One.	Inputs 12 & 13. Software selectable.
Jog, Hold, Stop	Three.	Software assignable to programmable inputs.
Active State	High on Power Up I/O ports TTL logic low.	High or low. Software selectable.
Outputs		
Type	Open collector with pull-up resistors.	Optically isolated. Open collector 5-15VDC, 25 milliamp maximum.
Fault	N/A.	Overtemp, under-voltage or under-current.
Programmable	Two. TTL logic low level-On.	Eight. Active high or low. Software selectable.
Encoder		
Channels	None.	Complimentary A & B channel in quadrature with index channel. Maximum input frequency rate of 256 kHz on A & B channel (prequadrature).
Dimensions	4" W x 4" H x 4-1/4" L	JDI-6M = 3" W x 9-1/2" H x 6-7/8" L JDI-8M* = 4-9/16" W x 9-1/2" H x 6-7/8" L *Heatsink added

62

Control Block Diagrams & Accessories

SP01 Switch Pack

Switch Pack
Motor
DC input

JDS Series

Linear Potentiometer
Motor
Mag. Switches (Limits)
115 VAC 50/60 Hz
JDS
(POM) Positioning Module
Speed Cycling Module
Timing Module

JDI Series

Motor
Encoder
Mag. Switches (Limits)
115 VAC 50/60 Hz
JDI Series
RS-232 Terminal (PC, HHT-232, etc.)

JBU Series

Encoder
Motor
Mag. Switches (Limits)
115 VAC or 230 VAC
JBU 100
or
*JBU-HT
*(Hand-held terminal) or PC
J1
RS-232 Terminal (PC, HHT-232, etc.)
5 VDC
115 VAC 50/60 Hz

HHT-232 (Hand-held programmable terminal)

4.125"
Display Screen
7.125"
Keypad
RS-232 Input Jack
3.57"
1.0"

SP01 & *SP01R Switch Pack

DC In (Power Source)
Momentary Switch (Hold-to-Run)
18 AWG (6' Length)
EXTEND
RETRACT
18 AWG (6' Length)
DC Out (Motor Leads)
2.125"
1.625"
4.00"

*For driving 90 VDC motors from a 115 VAC Input.

67

APPENDIX E

Crack Finding Program

```
/*****************************************************************
*                                                               *
*       File:           lss.c                                   *
*                                                               *
*       Date:           6/26/96                                 *
*                                                               *
*       Author:         Lisa Matsumoto                          *
*                                                               *
*       Function:       This program first locates the crack in the pavement after the *
*                       user positions the TMRR over the crack.  Once the starting *
*                       position of the crack has been located, the program tracks the *
*                       crack.  For this program, the motor resolution of the step motor *
*                       included in the linear positioning system is set to 400 steps per *
*                       revolution.  Be sure the SETUP file stored in the linear *
*                       positioner's indexer and the header file lss1.h is set accordingly. *
*                       Most of the functions were placed in seperate files and compiled *
*                       and then linked to for the library lss.lib to reduce the length of *
*                       this program.                           *
*                                                               *
*****************************************************************/

/*** header files ***/
#include "lss1.h"

// #define DEBUG
// #define SAMP_RATE

/*** function prototypes ***/
extern void wait_for_profile(void);
extern void s_cam_to_user(F_COOR *ptC, CALIBRATION8 near *ptCal);
extern void init_serial_ports(void);
extern void initalize(void);
extern void check_stop(int *run_status);
extern int rsready(void);
extern unsigned char rgetc(void);
extern void sendline(unsigned char *s);
extern void init_port(void);
extern void send_move_command(float offset);
extern double get_position(char s2[]);
int find_clean_crack(int counter, FILE *out);
float filter(float x[], float y[]);
float average(short a[], float tot_cells);
```

```c
void main(void)
{
    int type_of_crack, count = 0, counter = 1, i, j, c, jjj;
    int run_status = START_PROG, loop, crack_found = 0, crack_count;
    float tot_offset=0.0;
    char reply[10], buffer[80], s2[100], ch, *s, y;
    clock_t time1, time2, start_time, end_time;

#ifdef DEBUG
    FILE *out;
    out = fopen("out.out", "w");
#endif /* DEBUG */

    /* Initialize the computer, LaserVision system, and linear positioning system*/
    initialize();
    init_serial_ports();
    init_port();
    sendline("CALL SETUP");       /* Calls the preprogrammed initialization routine */

    /* Wait 35 seconds for initialization of linear positioner to be completed */
    time1 = clock();
    do
    {
        time2 = clock();
        printf("\r waiting for initialization to be completed");
    } while((time2 - time1) <35000);
    printf("\nINITIALIZATION IS COMPLETE!!\n");

    run_status = START_PROG;

#ifdef SAMP_RATE
    counter = 0;
    start_time = clock();
#endif /* SAMP_RATE */

    do
    {
        find_clean_crack(counter, out);
        counter++;
        check_stop(&run_status);
    } while(run_status != END_PROG);

#ifdef SAMP_RATE
```

```
        end_time = clock();
        printf("\ncounter = %d  ", counter);
        printf("sampling interval is %f\n", counter*1000.0/(end_time-start_time));
#endif /* SAMP_RATE */

        sendline("CALL QUIT");
        printf("end of program\n");
        fclose(out);
        clear_stdin();
}




/*****************************************************************
 *   This function determines whether or not a crack has been identified.  Identification   *
 *   of the crack is determined using the gradient method.                                  *
 *****************************************************************/
int find_clean_crack(int counter, FILE *out)
{
        F_COOR f_coor;
        int m, n, i, mincell, maxcell, med, jjj = 0;
        static int c = 0;
        float position[2], depth[2], ave, deriv[239], filt[240], offset;
        float width, max=0.0, min=0.0, x[3], y[2];
        static double e_pos = 0.0;
        char s2[100], ch, *s;

        /* reads a profile */
        wait_for_profile();

        /* obtains the average of the pixel values that compose the profile  */
        ave=average(address,240.0);

#ifdef DEBUG
        printf("average is %d\n", (int)ave);
        fprintf(out,"average is %d\n", (int)ave);
#endif /* DEBUG */

        /* Passing the data through a thresholding filter */
        if(address[0] < 340)
                address[0] = (int)ave;
        for(i = 1; i < 240; i++)
        {
                if((address[i]<340)||(address[i]>500))
```

```
            address[i] = address[i-1];
        if((abs(address[i-1]-address[i+1]))<=2 && (abs(address[i]-address[i-1]))>5)
            address[i]=address[i-1];
    }


/*   Filtering with Butterworth Filter (Cut-off frequency is 3 Hz and sampling      */
/*   rate is 33 Hz) and taking the derivative (or difference since we're dividing by  */
/*   one).                                                                            */
filt[0] = address[0];
filt[1] = filt[0];
deriv[0] = filt[0]-filt[1];
for(i = 2; i < 240; i++)
{
    x[0] = address[i-2];
    x[1] = address[i-1];
    x[2] = address[i];
    y[0] = filt[i-2];
    y[1] = filt[i-1];

    filt[i] = filter(x,y);
    deriv[i-1]=filt[i-1]-filt[i];
}


/*   Searching and locating the maximum negative spike   */
for(m = 15; m < 230; m++)
{
    if(deriv[m] > min)
    {
        min = deriv[m];
        mincell = m;
    }
}


/*   Searching and locating the maximum negative spike.   */
for(m = maxcell; m < 225; m++)
{
    if(deriv[m] < max)
    {
        max = deriv[m];
        maxcell = m;
    }
}
```

```
/* if the magnitudes of the largest positive and negative spikes are both less than */
/* 3.0, then there is no crack.                                                      */
if((fabs(min) < 3.0) && (max < 3.0))
{
        printf("\nNo crack found!!\n");
        return 1;
}
/* otherwise, there is a crack. */
else
{
        if(max > 0.0 && min == 0.0)
                mincell = 235;
#ifdef DEBUG
        printf("maxell is %d and mincell is %d\n", maxcell, mincell);
        fprintf(out,"max is %f @ %d and min is %f @ %d\n", max, maxcell, min,
                                                                        mincell);
#endif
        /*  Determining the location of the target by converting camera coordinates   */
        /*  to user coordinates.  The cell numbers are shifted back by 3 to           */
        /*  compensate for the phase shift caused by the Butterworth filtering.       */
        f_coor.line = (float)(maxcell);
        f_coor.pixel = (float)(address[maxcell]);
        s_cam_to_user(&f_coor, calib8);
        position[0]=f_coor.u;

        f_coor.line = (float)(mincell-5);
        f_coor.pixel = (float)(address[mincell-5]);
        s_cam_to_user(&f_coor, calib8);
        position[1]=f_coor.u;

        width=fabs(position[0]-position[1]);
        printf("width is %f mm\n", width);
        /* determines if the width of the crack is acceptable for sealing */
        if(width <3.2)
        {
            printf("crack too small\n");
            return 1;
        }
        else if (width > 50.0)
        {
            printf("crack too large\n");
            return 1; .
        }
```

```
            else
            {
                offset = (position[0]+position[1])/2;
                /*    If the offset is greater than 0.25 in, then move the linear positioner.   */
                if(fabs(offset) > 6.5)
                {
                    send_move_command(offset);
                    printf("offset: %.2fin\t", offset/25.4);
                }
                e_pos = -get_position(s2);
#ifdef DEBUG
                printf("position along lp: %lfin.\n", e_pos);
                fprintf(out,"offset: %.2fin, position along lp: %lfin.", offset/25.4, e_pos);
#endif /* DEBUG */
                return 0;
            }
        }
}


/* discrete realization for a recursive high pass filter */
float filter(float x[], float y[])
{
    float vo;

    vo = B1 * x[2] + B2 * x[1] + B3 * x[0] - A2 * y[1] - A3 * y[0];
    return vo;
}


/* returns the average of the array a */
float average(short a[], float tot_cells)
{
    float sum, ave;
    int h;

    sum = a[0];
    for(h = 1; h<tot_cells; h++)
        sum += a[h];
    ave = sum/tot_cells;
    return ave;
}
```

```
/********************************************************************
*                                                                  *
*       File:           lss1.h                                     *
*                                                                  *
*       Date:           6/26/96                                    *
*                                                                  *
*       Author:         Lisa Matsumoto                            *
*                                                                  *
*       Function:       This is a header file for the program lss1c.c. *
*                                                                  *
*                                                                  *
********************************************************************/

/*** header files ***/
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <signal.h>
#include <time.h>
#include <graph.h>
#include <bios.h>
#include <dos.h>
#include <conio.h>
#include <string.h>

/* MVS' header files */
#include "calib8.h"
#include "profile.h"
#include "register.h"
#include "private.h"
#include "filter.h"
#include "menu.h"

/*** symbolic constants and macros ***/
#define TOO_SMALL       1.0     /* width of crack in mm which is too small to seal */
#define TOO_BIG         50.0    /* width of crack in mm which is too large to seal */
#define N               25
#define NOT_FOUND       0
#define FOUND           1
#define YES             1
#define NO              0
#define INTENSITY       9       /* intensity of laser (1-9) */
#define START_PROG      1
```

```
#define END_PROG          0
#define FS_MM             101.6
#define NUM_PTS           nb_line_field   /* number of points in each scan line */
#define FS_BITS           126.0           /* full scale number of bits          */
#define SEND_DATA         26
#define NO_CRACK          1000.0
#define MAX_ERR           (FS_MM/2.4) /* Maximum error for saturation           */
#define MAX_NC_SAMPS      50          /* Maximum consecutive samples of         */
                                      /* no crack found before really sending   */
                                      /* the no crack signal to the robot.      */


#define COM1              0       /* com1 port assignment */
#define COM2              1       /* com2 port assignment */


/* Set crack profile filter constants (position filter) for Second Order Butterworth */
/* Filter.  Constants were found using Matlab (fc = 3 Hz, fs = 33 Hz). */
#define A1          1.0000
#define A2          -1.2172
#define A3          .4469
#define B1          .0574
#define B2          .1148
#define B3          .0574


#define TIME_FILTER
#define SATURATION_CHECK

/* Set time filter constants */
/* fc = 4 Hz, fs = 33, n = 2, r = 0.5 */
#define AT1         1.0000
#define AT2         -0.83473496972638
#define AT3         0.37065462937979
#define BT1         0.1265
#define BT2         0.2530
#define BT3         0.1265


/* fc = 1 Hz, fs = 33, n = 2, r = 0.5 */
/*
#define AT1         1.0000
#define AT2         -1.715154
#define AT3         0.763242
#define BT1         0.0113
#define BT2         0.0227
#define BT3         0.0113
```

```
*/

#define ESC_KEY      0x1b
#define CR           13
#define LF           10
#define TRUE         1
#define FALSE        0
#define BUFLEN       0x400      /* data buffer length */


/* When using COM2 to communicate with the linear positioner */
#define DATA         0x2f8          /* Data buffer register       */
#define IER          0x2f9          /* Interrupt Enable Register      */
#define IIR          0x2fa          /* Interrupt ID Register     */
#define LCR          0x2fb          /* Line Control Register     */
#define MCR          0x2fc          /* Modem Control Register      */
#define LSR          0x2fd          /* Line Status Register     */
#define MSR          0x2fe          /* Modem Status Register */


#define ONMSK        0xe7
#define OFFMSK       0x18
#define PICMSK       0x21           /* 8259 Mask Register     */
#define PICEOI       0x20           /* 8259 EOI Command     */
#define IRQ4         0x0c           /* COM1 interrupt vector */
#define IRQ3         0x0b           /* COM2 interrupt vector */


#define STEPS_PER_IN     800
#define BASE_10          10
```

```
/*****************************************************************
*                                                               *
*       File:           communi.c                               *
*                                                               *
*       Date:           6/26/96                                 *
*                                                               *
*       Author:         Lisa Matsumoto                          *
*                                                               *
*       Function:       Send_move_command converts the offset error obtained by the  *
*                       laser sensor into motor steps and then sends the "move"  *
*                       command to the linear positioner controller.  Get_position  *
*                       obtains the position of the carriage on the linear positioner in  *
*                       enoder pulses and then converts the pulses into distance.  This  *
*                       file was linked with other files and then compiled to form the  *
*                       library lss.lib.                        *
*                                                               *
*****************************************************************/

/*** header files ***/
#include "lss1.h"

/* #define DEBUG */

/*** function prototypes ***/
extern int rsready(void);
extern unsigned char rgetc(void);
extern void sendline(unsigned char *s);



/*****************************************************************
*  This function converts the offset obtained by the LaserVision System into the  *
*  equivalent number of motor steps and sends this value to the indexer.  *
*****************************************************************/
void send_move_command(float offset)
{
    int j;
    long int steps;
    char number_string[10], move_command[20] = "move ";

    steps = -(int)(offset/25.4*STEPS_PER_IN+50.0);
    ltoa(steps, number_string, BASE_10);
    for(j=0; j<strlen(number_string); j++)
    move_command[5+j] = number_string[j];
```

```
        move_command[5+j] = '\0';

#ifdef DEBUG
        printf("%s\n", move_command);
#endif
        sendline(move_command);
}



/*********************************************************************
 * This function obtains the encoder reading on the motor of the linear positioning  *
 * system and converts the encoder pulses to the position along the linear           *
 * positioner.  The position along the linear positioner is then returned to the main *
 * program.                                                                           *
 *********************************************************************/
double get_position(char s2[])
{
        char value[20], *s, s2[100], ch;
        int f, i, jjj = 0;
        static double pos;

        s = s2;

        sendline("VER E");

        while(1)
        {
            if (rsready())
            {
                jjj = 0;
                while( (ch=rgetc()) != CR )
                {
                    s2[jjj] = ch;
                    jjj++;
                }
            }

            if((s = strstr(s2, "E = ")) != NULL)
            {
                s += 4;
                i = 0;
                while((*s) != '\"')
                {
```

```
                value[i] = *s;
                s++;
                i++;
            }
            value[i] = '\0';
            pos = atof(value)/4000;
            break;
        }
    }
    return pos;
}
```

```
/*****************************************************************
*                                                               *
*       File:           initi.c                                 *
*                                                               *
*       Date:           6/26/96                                 *
*                                                               *
*       Author:         Debbie Krulewich                        *
*                                                               *
*       Function:       This file contains the functions that initialize the computer,   *
*                       LaserVision System, and the linear positioning system and some    *
*                       miscellaneous functions.  This file was linked with other files   *
*                       and then compiled to form the library lss.lib.                    *
*                                                               *
*****************************************************************/


/*** header files ***/
#include "lss1.h"

/*** function prototypes ***/
extern void (interrupt far *oldvect)();                 /* save original vector */


/*****************************************************************
* This function checks to see if any key on the keyboard has been hit.  If a key has  *
* been hit, then the program is to be terminated.                                     *
*****************************************************************/
void check_stop(int *run_status)
{
    if(kbhit())
        *run_status = END_PROG;
    else
        *run_status = START_PROG;
}


/*****************************************************************
* This function initializes the serial port COM 2 for communication between the       *
* the computer and the indexer of the linear positioning system.                      *
*****************************************************************/
void init_serial_ports(void)
{
    unsigned data2;
```

```
        data2 = (_COM_CHR8 | _COM_STOP1 | _COM_NOPARITY | _COM_9600);
        _bios_serialcom( _COM_INIT, COM2, data2);
}


void emergency_out(int sig)
{
    exit(0);
}



/******************************************************************
 *  This function lists the items that need to be executed at the completion of the    *
 *  program.                                                                            *
 ******************************************************************/
void last_call(void)
{
    outp(MCR, 0x00);                                /* disconnect phone line     */
    outp(IER, 0x00);                                /* interrupt disable         */
    outp(PICMSK, (inp(PICMSK) | OFFMSK));           /* restore INT mask bit of PIC */
    _dos_setvect(IRQ3, oldvect);                    /* restore the original vector */

    _restore_kb();                  //must be done, key board ISR has been changed
    p_int_reset();
    restore_vectors();
    u_reset_dir();                  //return to previous directory
    set_mode(3);
}



/******************************************************************
 *  This function initializes the LaserVision System.                                  *
 ******************************************************************/
void initialize(void)
{
    int i;

    if (signal(SIGINT, emergency_out) == SIG_ERR)    /*trap ^C*/
    {
        perror("signal failed");
        exit(0);
    }
    atexit(last_call); /* the system will call last_call when the program terminates */
```

```
p_trap_kb();              /*save actual key board fct so on return we restore it*/
u_set_dir();              /*See II.3 Definition of proper DOS environment*/
set_LPB_board(0);         /*read if present descriptions of the LPB board*/
p_int_reset();            /*resets interrupts 1-6 enable bit in CTRL1 rgister */
p_int_init();             /*must be done once only*/
global_init();            /*read camera parameters*/
p_init_all();             /* initialize board */
error_msg("End of initialization, ready to acquire continuously a profile");
set_laser(INTENSITY);     /* set laser intensity */
}
```

```
/*****************************************************************
*                                                               *
*       File:           rs232.c                                 *
*                                                               *
*       Date:           6/26/96                                 *
*                                                               *
*       Author:         Daehie Hong                             *
*                       Scott Winters                           *
*                                                               *
*       Function:       These programs allow communication between the linear    *
*                       positioner's controller and the computer through the serial port  *
*                       connection.  This file was linked with other files and then        *
*                       compiled to form the library lss.lib.                    *
*                                                               *
*****************************************************************/


/*** header files ***/
#include "lss1.h"

unsigned char ibuffer[BUFLEN];                  /* input data buffer */
unsigned int ifront = 0, irear = 0;             /* data pointers     */

void (interrupt far *oldvect)();                /* save original vector */


/*  Interrupt Service Routine                                    */
/*  This routine is interrupted whenever Data Ready and reads data from    */
/*  data port and save it to the buffer.                         */
void interrupt far intsr_sc(void)
{
    irear %= BUFLEN;
    ibuffer[irear++] = (char) inp(DATA);
    outp(PICEOI,0x20);
}


/*      Checks if the buffer has any input data.      */
int rsready(void)
{
    return((ifront != irear) ? TRUE : FALSE);
}
```

```c
/*      read one character from buffer     */
unsigned char rsin(void)
{
    ifront %= BUFLEN;
    return (ibuffer[ifront++]);
}



/*   Reads one char if the buffer has input characters, and returns NULL if there    */
/*   is no input char after waiting for some time.                                    */
unsigned char rgetc(void)
{
    int tzcount = 0;
    int timeout = 0;

    while(!rsready())
    {
        tzcount ++;
        if (tzcount > 10000)
        {
            timeout ++;
            break;
        }
    }
    if (timeout)
        return(0);
    else
        return(rsin());
}



void rsout(unsigned char ch)
/*      send one char through communication line     */
{
    while( !(inp(LSR) & 0x20) );        /* wait until ready to send */
    outp(DATA, ch);                      /* send a char to data port */
}


#define DELAY 1000
```

```c
/*      Sends a string to end with NULL char.    */
void sendline(unsigned char *s)
{
    int i=0;
    while(*s)
    {
        rsout(*s++);
        for( i; i<DELAY; i++);
    }
    i=0;
    rsout(CR);
    for ( i; i<DELAY; i++);
}


/*      Initializes communication port and interrupt enable register.   */
void init_port(void)
{
    oldvect = _dos_getvect(IRQ3);           /*      save old vector      */
    _dos_setvect(IRQ3, intsr_sc);           /*      set a new vector      */

    outp(MCR, 0x0b);
    outp(IER, 0x01);                        /* set bit 0 of interrupt enable register  */
    /* so, interrupt can happen when data ready*/
    outp(PICMSK, (inp(PICMSK) & (ONMSK)));
    /*      set INT mask bit of 8259 PIC        */
}
```

SETUP

This program is used when the motor resolution is set to 400 steps/rev. The file *setup400* contains SETHOME that goes with this program.

MR = 400	^ motor resolution = 400 steps/rev
ER = 4000	^ encoder resolution = 4000 pulses/rev
UR = 360	^ user resolution = 360 degrees/rev
A = 1000	^ accelerate in 1 second
B = 800	^ base speed = 800 steps/sec
M = 1400	^ maximum velocity = 1400 steps/sec
D = 1000	^ decelerate in 1 second
Q = 0	^ no time delay to return drive to full power
H = 3200	^ high speed limit = 3200 steps/sec
S = 2000	^ reduce current to no-power 20 seconds after motion is complete
MOVE 0	^ calculate the ramp table without moving
CALL SETHOME	^ call the program "SETHOME"

SETHOME

This program is used when the motor resolution is set to 400 steps/rev. This program is a part of the file *setup400*.

LA = 00000000	^ motor will be stopped without decelerating when a STOP is encountered
OF = 00000001	^ activate pause processing mode
- RUN	^ run in the CCW direction
(START) UNTIL 15 START	^ wait until the CCW limit switch becomes active
P = 0	^ set current position to 0
MUNITS	^ select motor units
EF = 00000000	^ all encoder function parameters is inactive
WAIT 100	^ wait for 1 second
MOVE 8586	^ move 8586 steps
E = 0	^ set current encoder position to 0
P = 0	^ set current position to 0
UP = 0	^ set current user position to 0
WAIT 100	^ wait for 1 second
MOVE 7900	^ move 7900 steps
OF = 00000000	^ activate continuous processing mode