Development of a Machine Vision Based Oyster Meat Sorter

by

Maria B. Koslav

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Agricultural Engineering

APPROVED:

Dr. Richard K. Byler, Chair

Dr. Kenneth C. Diehl, Co-Chair

Dr. Charles F. Reinholtz

November, 1989

Blacksburg, Virginia

Machine Vision Based Oyster Meat Sorter

by

Maria B. Koslav

Committee Chairman: Richard K. Byler

Agricultural Engineering

(ABSTRACT)

Oyster meats are currently sorted by hand using volume as the sorting parameter. Hand grading is inaccurate, time consuming and costly. Previous research on physical properties of oyster meats showed a high correlation between projected area of oyster meats and their volume thus allowing the use of projected area measurements as a sorting criterion. A machine vision based oyster meat sorting machine was developed to mechanize the sorting process. The machine consists of a dark conveyor belt transporting singulated oysters through a grading station and then along a row of fast acting water jet valves which separates the stream of oysters into 3 classes. The vision system consists of a monochrome television camera, flash light illumination to "freeze" the images, a digitizer/transmitter and a Personal Computer as an image processing unit. Software synchronizes the flash light and digitization of images and calculates projected area of each meat using the planimeter method. The grading results are sent to a valve control board which actuates the spray valves. The sorting rate is 37 oyster meats/min with a sorting accuracy of 87.5%. A description of the design work, adjustment and calibration procedures and a final sorting test is included.

Acknowledgements

I express my deep appreciation for the guidance and friendship offered by my major advisor and Committee Chairman, Dr. Richard K. Byler, during my graduate studies. Byler's efforts in writing the spray system control software were invaluable in the fruition of my work. I also appreciate the support and helpful advice given my by my Committee Co-chairman Dr. Kenneth Ç. Diehl, and Committee member Dr. Charles F. Reinholtz.

Special thanks are extended to and for their valuable suggestions and the realization of my design work in steel at Virginia Tech's Agricultural Engineering shop. In addition, my friend devoted many hours to editing my writing and providing the photographic work presented in this thesis.

Finally, I am most thankful to my family members in Germany. Without their support, my graduate work in America would not have been possible.

Table of Contents

List of Illustrations

List of Tables

1. Introduction

Fresh oyster meat is not only a delicious seafood, oyster harvesting and processing are an important source of income for the seafood industry in the Gulf, Atlantic, and Pacific coast regions of the United States. However, in recent years the oyster industry has faced difficult times. The oyster beds in the lower Chesapeake Bay in Virginia have suffered from severe losses because of parasite infestations and pollution. At the same time the entire industry suffers due to increasing labor costs and decreasing availability of skilled workers. Efforts to mechanize or even automate post-harvest processing would help preserve the East Coast oyster industry and strengthen it against future challenges.

Manual labor requirements for oyster processing are high, because each oyster must be manually shucked (the act of removing the meat from the shell), cleaned, graded, and packed after harvest. Shucking is hard, tedious work which requires precise motions with hammer and knife to open the shell and cut the abductor muscle without damaging the oyster meat (Figure 1). Many attempts have been made to mechanize

Figure 1 Oyster Shucking Operation

the shucking process, so far without any commercial success. (Smith, 1971; Wheaton, 1974; Wheaton and Story, 1974, Wheaton, 1985; Tojeiro and Wheaton, 1987; Chen and Wheaton, 1987).

No significant efforts have been made to automate the grading and sorting process. Automated sorting of oyster meats would lower labor requirements, facilitate machine packing, increase the number of oysters shucked, and provide more consistent and reliable grading than that obtained by the human eye. Automation of grading and sorting would establish a more common basis for trade and would therefore benefit both buyer and seller of oyster meats. Figure 2 shows the labor involved in grading and packing in a large operation. At the present time, human graders sort fresh oyster meats on the basis of volume following Food and Drug Administration standards (FDA, 1988), which are based on counts per gallon. Two different standards exists one for the Eastern (Crassostrea virginica) and the other for the Pacific (Crassostrea gigas) oyster species. Table 1 contains those grade standards along with the calculated average volume for each oyster based on the counts per gallon. These calculated volumes are based on the assumption that a gallon container is only filled with meats. Those standards are not followed closely in practice. Oyster are usually harvested when the majority of the bed population reaches *standard* or *select* size having the consequence that packers usually sort only into three categories, *standard*, *select* and *count*, which refer to the *very small* category (Spady, 1988; Hayes, 1988).

An average processing plant on the Atlantic coast employs 15 to 30 shuckers, which

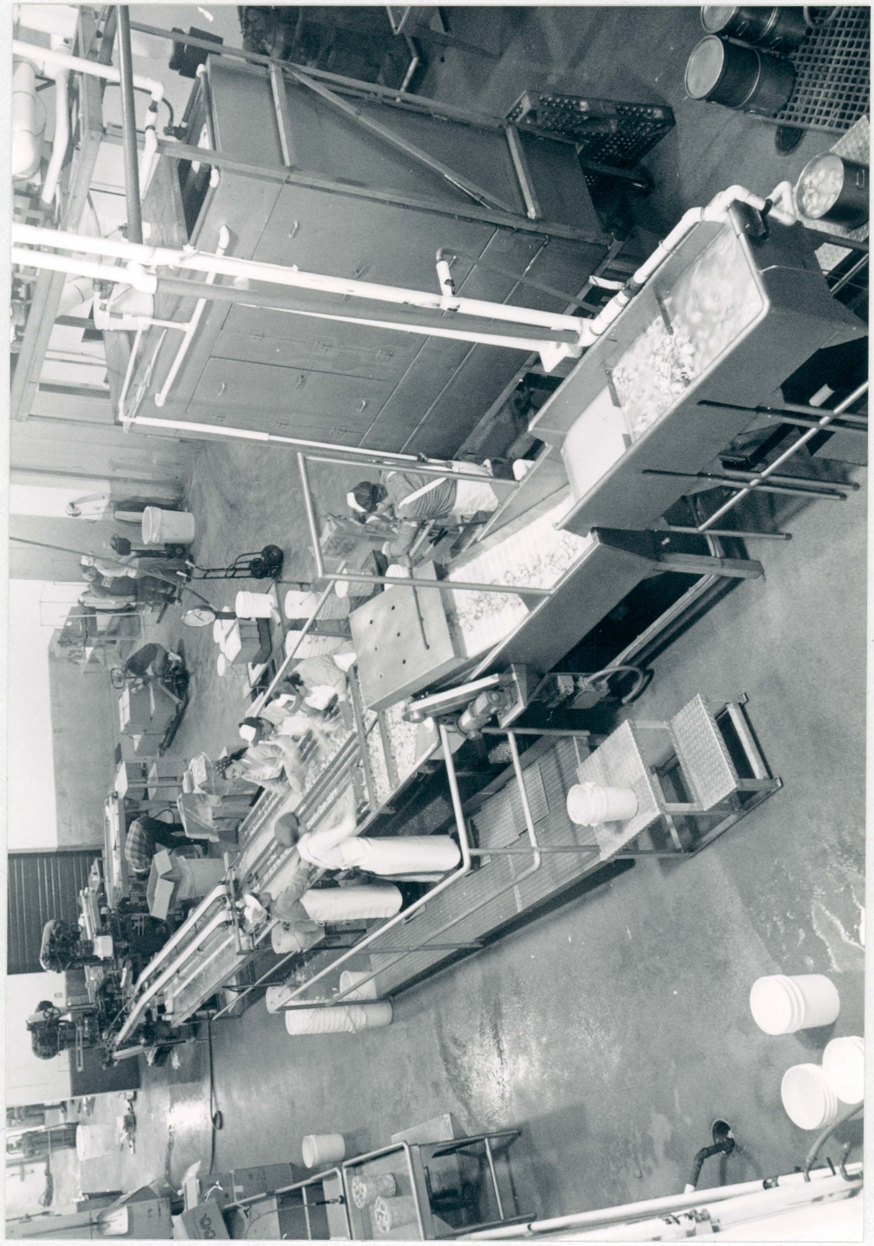


Figure 2 Grading and Packing of Oyster Meats

simultaneously do the grading, and 1 or 2 packers. Oyster shucking operations at the Pacific Coast are most often larger, employing between 50 and 100 people. A shucker averages between 500 to 600 oysters/hour. One grader can process the output of 10 shuckers, or 5,000 to 6,000 oyster meats/hour. One sorting machine with a capacity of 18,000 oyster meats per hour or 5 oysters/second would be sufficient for a medium size East Coast operation.

Automation of oyster meat grading was investigated by Awa (1988). He used a machine vision system to measure projected area of oyster meats. He found a high correlation ($r = 0.91$) between projected area of the meats and their volume which permitted the use of projected area as a sorting criterion. His device graded 5 oyster meats per second with a measurement error of 3.0 cm^2. His results (which will be discussed in more detail in Section 3.3) proved the feasibility of a machine vision system to grade raw oyster meat. Coupling such a system to a sorting device would provide a commercially viable and highly valuable machine for the oyster processing industry.

Table 1 Size Grade Standards for Oyster Meats

Eastern Oyster Meats

Size	Counts (per gal)	Volume (cm^3)
very small	>500	<7.57
standard	301-500	7.57-12.6
select	211-300	12.6-18.0
extra select	160-210	18.0-23.7
extra large	<160	>23.7

Western Oyster Meats

Size	Counts (per gal)	Volume (cm^3)
extra small	>144	<26.3
Small	97-144	26.3-39.0
Medium	65-96	39.0-58.2
Large	<64	>59.2

2. Objectives

The overall objective of this study was to develop a prototype microprocessor-controlled grading and sorting machine for raw oyster meats that operates at a rate of 5 oyster meats/s sorting oyster meats into 3 different size categories.

Specific objectives were:

1. to develop an electronic system to optically measure the projected area of oyster meats at the desired grading rate,
2. to develop an oyster meat conveying device that permits projected area measurement and sorting the oyster meats into designated containers,
3. to determine the relationship between pixel number and projected area of the oyster meats, and
4. to determine the device's sorting accuracy.

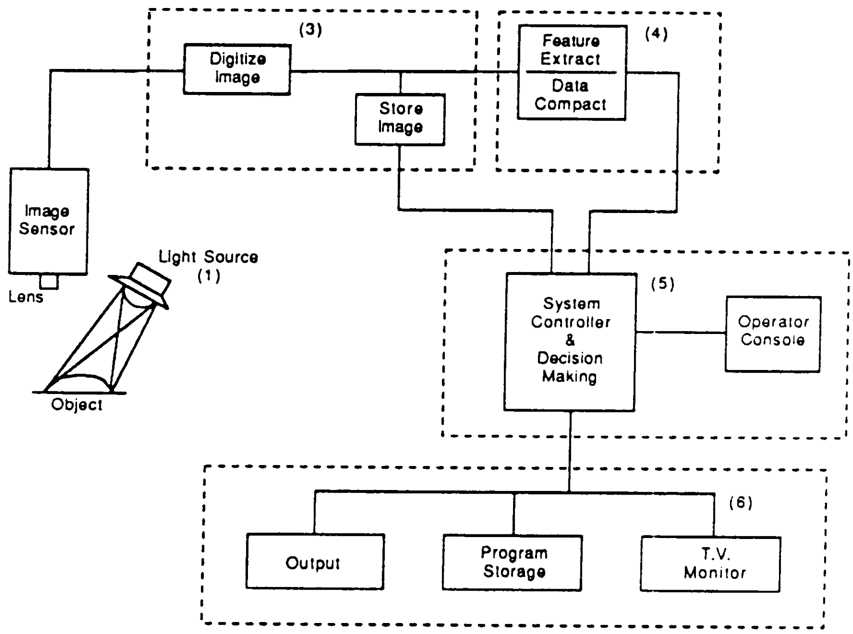
3. Literature Review

The heart of the proposed oyster sorter is its machine vision grading system, as suggested through previous research on the same subject. The literature investigation for this project ranged from machine vision applications in industrial manufacturing process (where most the advanced vision technology is used) to machine vision applications in agriculture. It was necessary to study literature on applications in industry to learn about object lighting and viewing techniques, availability of sensors, how to transform optical images into digital form, and how to manipulate images to extract object properties. Because of the substantial cost decrease and increased power of microcomputers, many different applications for vision systems in agriculture have emerged in recent years. A selection of different agricultural applications is listed in the second part of this chapter. This chapter also contains a summary of research in oyster meat sorting mechanization.

3.1 Machine Vision

Machine vision is defined by the Machine Vision Association (MVA) of the Society of Manufacturing Engineers (SME) as "the use of devices for optical, non-contact sensing to automatically receive and interpret an image of a real scene in order to obtain information and/or control processes or machines" (Zuech, 1988). Basic machine vision systems generally include the components shown in Figure 3. A sensor captures electromagnetic energy (typically in the visible spectrum) from a scene and converts it to an analog electronic signal or optical image. Optical lenses sometimes supplemented by filters, couple the image to the sensor. An illumination system specifically designed for the subjects must usually be employed because ambient light is usually not bright and consistent enough to produce images with specific content. After digitizing and temporary storage the computer extracts data from the image -- often first enhancing or otherwise processing the data -- and compares the data with previously developed standards. The computer sends the results to the system, usually in form of an electronic signal through communication channels. Machine vision involves automatic image interpretation, process control, quality control, machine control, and robot control. The system operator can usually interface to the vision system through terminals, light pens, etc. while observing the image on a display screen (Zuech, 1988; Faugeras, 1983).

Compared with human eye-brain capacity current machine vision systems are primitive. The range of objects that can be viewed, speed of interpretation, and susceptibility to



GENERAL VISION SYSTEM BLOCK DIAGRAM

Figure 1.20 Block diagram of basic machine vision system.

Figure 3 Block Diagram of Basic Machine Vision System (Zuech, 1988) lighting problems, including minor variations in texture and reflectance of objects, are examples of limitations of current technology. Compared to automated inspection, human inspection is much simpler and faster to set up, and human versatility and judgement make strict and detailed specification of the product requirements and tolerances unnecessary. Conversely, machine vision has clear advantages when it comes to consistency in inspection results, in capacity to keep up with high line speeds, and being unaffected by fatigue and task monotony. People have a limited attention span which makes them susceptible to distraction. People are also inconsistent, with individuals often exhibiting varying sensitivities during the course of a day, from day to day, and from person to person (Cielo, 1988).

3.1.1 Lighting and Optical Techniques for Image Enhancement

Lighting can either enhance or obscure features to be detected. The first step of any machine vision application is to properly illuminate the object to render it detectable by the sensor. Poorly applied light can produce glare (which may saturate the sensor), low contrast, or picture non-uniformity making inspection difficult. Zuech (1988) describes the lighting objectives for machine vision system as follows:

- Optimize the contrast associated with the condition to detect.
- Normalize any variances due to ambient conditions.
- Simplify image processing, and therefore computing power required.

Different lighting techniques have been developed for various objects depending on their geometric properties, specularity, texture, color, the background, and data to be extracted from the image. Various techniques employ incandescent light bulbs, fluorescent tubes, discharge tubes, quartz halogen bulbs, strobe tubes, strobe tubes, light emitting diodes, lasers, and fiber optic light sources combined with reflectors, filters, and lenses to direct, condense, collimate, diffuse, or polarize the light. Batchelor et al. (1985) presents a detailed description of 63 lighting and viewing methods from which the most common principles of operation are described below and in Figure 4:

- **Backlighting.** When diffusely backlit, the profile of a thin, opaque object is sharply delineated and is used to extract geometric parameters from the picture. Collimated light is necessary when viewing a thick object.
- **Diffuse front lighting.** This is the typical approach for a high-contrast object, such as a white feature against a dark background. The field seen by the camera must be evenly illuminated.
- **Directional lighting.** This technique is particularly used for the inspection of surfaces, where shadows indicate edges, textures, or other surface features. Light used is usually a point source.
- **Structured lighting.** Points, stripes, or grids of light are projected onto an object. The angular projection of the object profile is then seen by the sensor. A powerful method is to spread a laser beam to a line scanning the profile of an object.

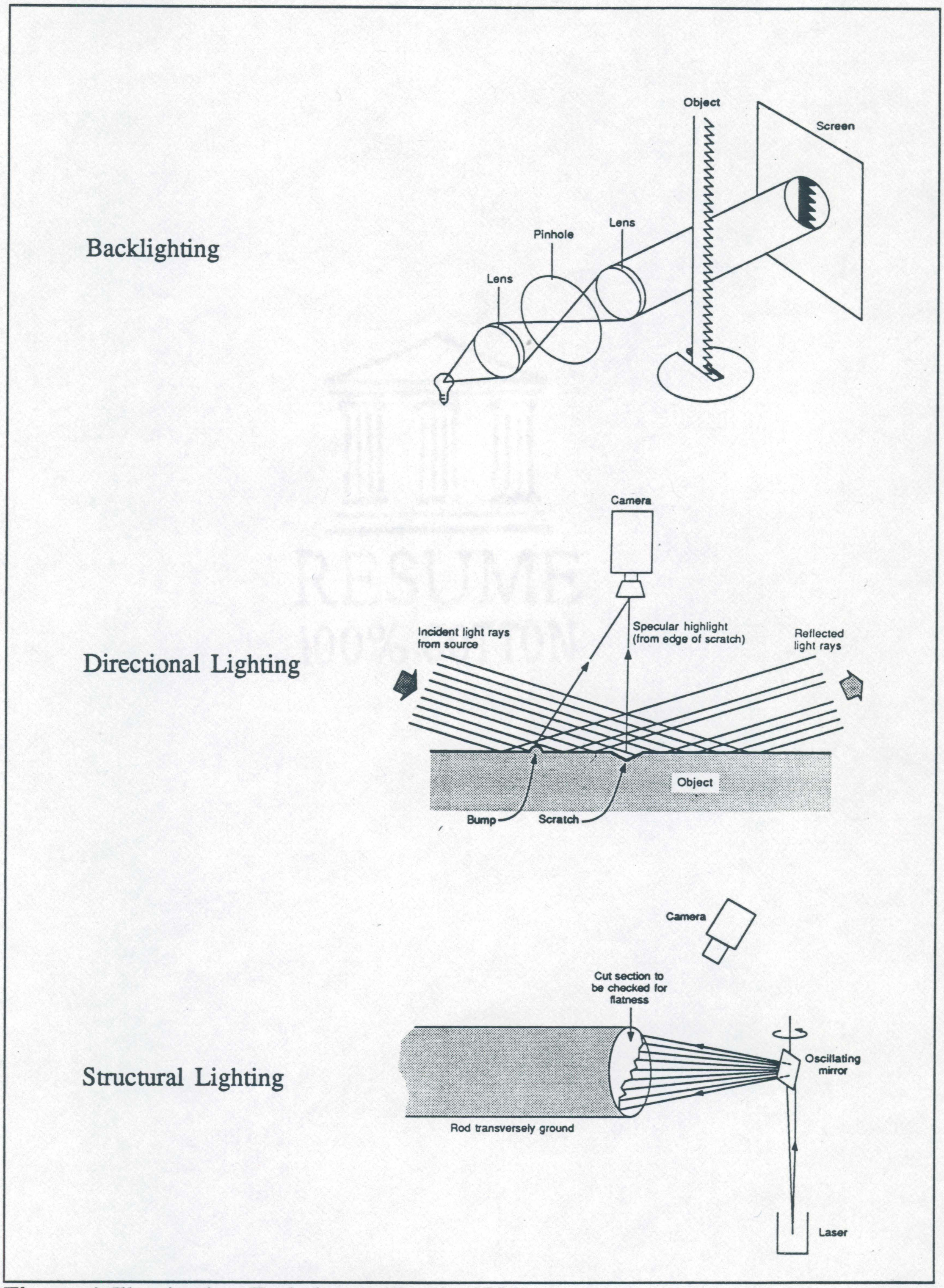


Figure 4 Illumination Techniques (Zuech, 1988)

Filters protect the sensor from unwanted wavelengths. Polarized light reduces specular reflections and glints of objects or backgrounds. More information about polarized light is given in Section 4.3.3.

3.1.2 Image Acquisition

Lapidus and Englander (1985) describe the hardware which converts an optical image into a series of numbers. It consists of three elements:

- Lens
- Sensor (camera)
- Digitizing Equipment

The lens forms an image on the sensor's light sensitive surface which is converted to a matrix of numbers by a separate digitizing element. Each number, or picture element (pixel) of that matrix represents the relative light intensity at that point in the original optical picture. The variation of intensities between black and saturation of the sensor are called gray scales. The number of gray scales usually varies most often between two (binary image) and 256 gray levels depending on the sensing device and digitizer. The gray scale information is sent to a host computer for processing.

The lens forms an image on the sensor and it can be a major source of error in a system because of various lens aberrations. Most important for machine vision systems are geometric distortion and non-uniform light intensity response. Geometric distortion means that the image is not the same shape as the object. A square, for example might be reproduced with inward curving sides (pincushion distortion) or outward curving sides (barrel distortion). The cause of distortion is the greater angle that rays from the corners of the square form with the axis of the lens than rays from the sides of the square. This can cause measurement errors up to 3% when an objects spans from the center to the edge of the field of view (Lapidus and Englander, 1985). Zoom lenses suffer in general more distortion than lenses with fixed focal length. Non-uniform response of lenses mean that a uniform sheet of light will not be projected with the same uniformity of intensity onto the camera sensor array. There tends to be gradual fall-off of light from the center to the edge of the image plane sometimes exceeding 2:1. This fall-off must be taken into account when using binary images. One threshold might work near the center of the image but a different one might be required at the edge. Fall-offs can sometimes be compensated for during image processing if their magnitude is known. For the choice of lenses Lapidus and Englander (1985) recommend "as a rule of thumb that the more glass they contain and the more they cost, they are less likely to generate serious imaging distortions".

Sensors in machine vision systems are usually phosphor-coated vacuum tubes or solid state sensors. Vacuum tube cameras have photosensitive targets which are scanned by an electron beam. A further description of vacuum tube cameras can be found in Section 4.3.2. Sources of error for vacuum tube cameras are non-linearity, drift, and non-uniformity of target coating. Non-linearity appears as a change of size of the pixel at different points in the image. A typical value for non-linearity is 3% which also needs to be compensated electronically.

Machine vision systems also employ solid-state sensors to capture image data. They use an array of light sensitive semiconductor devices integrated into a single chip. This type of camera directly digitizes the image. All solid-state cameras provide spatial stability because of the fixed arrangement of the photosensitive sites. Solid-state cameras are essentially free of drift and non-linearity but they vary from element to element in light intensity response. This can cause "dead zones" or blemishes in a picture (Zuech, 1988).

Other devices similar to solid-state sensors are linear solid-state arrays, either charge-coupled devices (CCD) or photo diodes. These provide single lines of video in high spatial detail in relatively short periods of time. Linear array cameras are frequently used in a fixed arrangement or to capture information as an object passes the array. Two dimensional information can be detected in a manner analogous to a train passing a barn as if viewing it through slats in the barn. Similarly. a linear array can capture the two-dimensional information of a moving object. The object motion must be repeatable, or travel speed well regulated, to obtain repeatable two-dimensional images from the buildup of independent linear images (Lapidus and Englander, 1985).

The digitizing element converts the analog voltage produced by the camera into a series of numbers which the computer can analyze. A comparator circuit converts an input voltage exceeding a certain threshold into a 1 for a binary system while voltages below that threshold are converted to 0. A circuit called a digital converter transforms the input voltages into numbers for a gray scale systems (see section 4.3.2).

3.1.3 Image Processing

After the image is in digital form accessible to a computer it is subject to image processing. Gonzalez et al. (1986) divide image processing into five areas: enhancement, restoration, compression, segmentation, and description.

Image enhancement deals with improving pictorial information for human interpretation and machine perception. Histogram equalization is a typical enhancement approach in the spatial domain. This creates an intensity transformation function that results in a flat histogram and increased contrast of the picture. Enhancement in the frequency domain is based on filtering, low pass filtering for smoothing and high pass filtering for sharpening the picture. Operations in spatial and frequency domain can be combined to enhance a picture. Image restoration is concerned with reconstructing or recovering an image that has been degraded. This is usually done by using *a priori* knowledge about the nature of the image degradation. Restoration techniques are oriented toward modeling the degradation and applying an inverse process to recover the original picture. Image compression techniques are useful whenever reducing storage or transmission time requirements are important. Compression techniques are either reversible or irreversible. Average compression ratios of 3:1 are common with reversible compression. Most common reversible compression methods are run-length, Huffman, and shift codes. Irreversible compression achieves compression ratios up to 37:1 but loses image detail in a reconstructed image. Segmentation is the process that subdivides an image into objects or regions of interest. The most widely used method is thresholding. Two reasons exist for this. First, thresholding is fast and easy to implement in hardware or software. Second, when lighting is sufficiently controlled, an image lends itself to a variable or constant threshold to extract an object. Edge detection is another method of choice to separate an object in an image. Description in image processing is to extract features from an object for the purpose of recognition. Those features are usually independent of location and orientation, and contain enough discriminatory information to differentiate one object from another. Descriptors are generally based on shape and amplitude (e.g. gray scale intensity) information and they are either globally or boundary oriented. Global descriptors include principal axis analysis, two and three-dimensional moment invariants, texture, perimeter, and area of objects. Among the boundary-oriented descriptors Fourier, and "tree" descriptors are well known.

Once the desired information is extracted from the picture the vision system has to classify the investigated object. The system compares picture information with a set of predefined standards, then classifies the object according to those standards. In a process control system the final classification decision will initialize appropriate actions of the machine controlled. Because of heavy computational load through image processing on the main processor manufacturers offer high speed preprocessors and coprocessors. These modules which include pre-programmed input look-up tables and array processors, perform common image processing functions in hardware to enable real-time operations (Goshorn, 1983).

3.2 Machine Vision Applications in Agriculture

Machine vision is firmly established in manufacturing environment for inspection and guidance tasks. The accuracy and speed of vision systems are critically affected by features selected for evaluation, as well as software and hardware design. Proper feature selection can retrieve representative information, simplify vision algorithms and increase the throughput. Manufactured parts are often uniform in shape, size and color simplifying the feature selection problem. Biological materials exhibit a much larger variation in appearance and color, making feature selection more complex. In recent years imaging technology has become less expensive through the emergence of powerful microcomputers and specialized hardware. And finally, recent progress has resulted in more robust and flexible vision systems with easily available off-the-shelf components. This has inspired a broad field of research application in agriculture. Machine vision systems in agriculture are used principally for image analysis, robotic guidance, and inspection (Kranzler, 1984).

The use of machine vision to recognize physical properties of seeds and small fruits was investigated by several researchers. Gunasekaran et al. (1988) detected stress cracks in corn kernels using backlighting, with a 256 x 256 pixel solid state camera using 64 gray levels. The backlighting was limited to a 2.4 mm hole in a black pad which contains the kernel when a picture is taken. Ninety percent of all examined kernels were classified correctly.

Berlage et al. (1988) searched for a new method to distinguish between diploid and tetraploid ryegrass seeds to replace the chromosome counting procedure under a microscope. They used a 64 gray level solid state camera with a resolution of 238 lines and 256 columns. A 38 mm fixed focus macro bellows lens provided a 7 by 9 mm field of view for the camera. The object stand was lit by a fiber optic ring light to eliminate shadows. The seeds were placed on a small black rubber pad with the rachilla side faced down. The pictures were analyzed for length, width, area, perimeter, reflectance, and several other parameters. A multivariate regression model was developed with one set of seeds. The model was verified in another set of seeds consisting of six tetraploid and two diploid varieties. Eighty-five percent of all seeds could be classified correctly.

Delwiche et al. (1988) developed a system to detect prune surface defects such as mold, scab and cracks. The defects lead to rejection by human inspectors before packaging. They used a 256 x 1 pixel line-scan camera with 16 mm focal length lens. The inspection scene was lit through two 100 W incandescent light bulbs in an illumination chamber. The chamber was painted white inside to provide shadow free diffuse light. A conveyor belt moved the prunes through the camera field of view at a grading speed of 20 fruits/s. The pictures were analyzed by a microcomputer based on the discontinuity and similarity of prune surface gray-level values because defective prune images showed higher spatial frequency components. After global thresholding which separated the prune from the background, defect edges were detected by gradient operations on each row. If the row gradient exceeded a certain threshold that row was marked defect. The prune was rejected if it had too many defect rows. Errors in classifying defect fruits between no error and 4.6% misclassification.

Robotic vision applications are still largely in the research stage but several concepts have been tested for distinguishing fruits from background to guide robotic arms for automated picking. One attempt to locate fruits in a natural scene was made by Whittaker et al. (1987). He detected green tomatoes on the basis of Hough's circular transform. The algorithm locates round objects even when they are partially covered. 37 recorded video pictures of tomato plants with fruits were investigated. The algorithm detected 76% of all tomatoes in those pictures.

Site and Delwiche (1988) investigated the usefulness of a computer vision system as a sensor for robotic peach and apple harvesting. The sensor was a solid state camera with a resolution of 128 rows by 128 columns and 256 gray levels. The frame grabber allowed a digitization and transfer of the picture in the computers memory in 66 ms. Pictures were taken in orchards under the light of 500 W floodlamps at daytime and at night. Pictures with enhanced contrast between background and fruit were obtained with a 630 - 670 nm bandpass optical filter. Non-fruit objects were removed from the picture with a percentage histogram converting the image into a binary image. Then the image was smoothed with a low pass filter and the objects distinguished from each other. Classification parameter for fruit and non-fruit objects were perimeter, perimeter/area ratio and the ratio of minimum to maximum moment of inertia. Eighty-nine percent of the peaches were identified in night time pictures and 79% in daytime. Eighty-eight percent of apples were identified. The lower identification rate in daytime was due to reflections of sunlight and varying light intensity (cloud cover).

There is a variety of commercialized systems to sort raw or processed food materials on the market. One of the earliest large scale automatic systems was a cucumber sorter. The machine consists of an illumination source, photo diode line sensor, an image processor and a multi-row tilting tray conveyor. Cucumbers are sorted into three grades according to shape and five sizes according to length at a rate of 10 cucumbers/s (Nakahara et al., 1979).

Marchant et al. (1988) developed in connection with Loctronic Graders Ltd., Danbury, UK, a high speed potato sorter. Potatoes were sorted for baking purposes. In order to be of baking quality potatoes must be of certain minimal size both overall in their three principal axes and they should be of a prolate spheroid shape. The working sorter consists of a 60 cm wide roller conveyor belt with the rollers linked to a plastic chain. When the rollers pass the camera field of view they are in contact with a stationary bed which causes them to rotate and also rotate the potatoes between them. The camera observes the rows of rotating potatoes. No singulation is necessary because the vision system can distinguish between touching potatoes. When the potatoes leave the field of vision the computer instructs the handling mechanism based on data from the image analysis. The handling mechanism consists of a row of pneumatically operated fingers deflecting the potatoes to one of two bins. The vision system consists of a closed circuit TV camera. A shaft encoder on a belt pulley provides the trigger signal to grab a frame making it independent from the belt speed. All the information is extracted from the detected outline of a potato which is repeated three times. The main microprocessor controls the action of the frame grabber, a chain encoder which finds the boundary data and three slave processors which keep track of each potato. Finally the main processor passes the grading results to a separate computer controlling the separation device. The system graded 20 potatoes/s and is expected to grade 40 potatoes/s in future.

Somers and Frank (1987) from Campbell Soups, Inc. in Toronto report the use of two optical sorters for randomly diced potatoes and carrots. The vision system employed for quality control detects off-color spots on the diced vegetables. A vibratory conveyor orients the vegetable into a single layer feeding the sorting machine conveyor belt. The vegetable pieces are inspected from 3 different angle at a conveyor speed of 1.77 m/s. When a reject is detected the system controller activates one or more air jet nozzles to eject the bad product to a take-away conveyor. Each machine is equipped with 64 air jets. Controlling individual air jets for more precise timing enables the system to eject fewer than one good piece with each defective piece. The sorter inspects 6,800 kg vegetable per hour.

3.3 Oyster Physical Properties and Processing Automation

Physical properties of materials are important in determining equipment to handle or process the material. A physical properties study for oyster meats was conducted by Diehl et al (1988). They measured height, weight, volume, and projected area of raw oyster meats in order to find a suitable and a more easily measured grading parameter than volume, which is the presently used grading parameter. Several hundred oysters (446) from 3 different locations, namely Virginia's Atlantic coast, the James River, and the Gulf of Mexico were measured. All oyster were Crassostrea virginica, eastern oysters. The measurements were taken with oysters lying flat on a glass plate, but no effort was made to spread out the gills or any other part of the oysters. This was considered to be similar to the way the oyster would lie during processing. The projected area was measured with a computer vision system and with a digital planimeter from a photographic print to have an independent measurement of area. Approximately 50 to 75% of those oysters were incorrectly graded outside of their designated category based on the standards of identity.

A very high correlation coefficient was found for the volume/area relationship (0.91) and for the Volume/weight relationship (0.98). This indicates that both parameters are good grading parameters, weight being better than projected area. Height and volume had a low correlation coefficient of 0.61, excluding it from being a valuable grading parameter (Awa, 1988). Diehl et al. (1988) developed relationships between volume and area, and volume and weight, which took the form linear slope intercept equations:

$$V = -0.61 + 0.00623 * A \qquad [1]$$

$$V = 0.99 + 0.886 * W \qquad [2]$$

where:

V = Volume [cm^3]

A = Projected Area [mm^2]

W = Weight [g]

Confidence intervals for predicted values at 95% confidence level were $\pm$1.7 cm^3 for prediction using weight and $\pm$3.9 cm^3 for predicted values from projected area.

Awa et al. (1988) developed a machine vision grader for oyster meats based on the results of the physical properties study. The system consisted of 1 m diameter, rotating disc as oyster conveying device. It was made from plexiglas and driven through a variable speed gearmotor. Single oyster meats were placed 0.255 m away from the rotation axis and spaced 18° apart. They moved through the camera field of view at a speed of 17 RPM which was the maximum rotational speed possible without propelling the meats from the disc due to centrifugal forces. The camera used was a *IS32A* Dynamic Random Access read/write Memory (DRAM) chip connected with a 16 mm lens. The chip consists of two arrays of pixels, 1.09 mm wide and 5.504 mm long. Each field consists of 128 by 256 pixel. To save processing time only alternate pixels from even rows (64 x 128) in one field were used. The chosen field of view at a 1 m lens-object distance was 5 cm by 26 cm resulting in a resolution of 1.6 mm^2/pixel. The digital data were transferred to the microcomputer at a baudrate of 153,600 bits per second using 67 ms for transfer. The oysters were backlit through diffused light from twelve 60 W incandescent lightbulbs underneath the disc. Incandescent light was chosen because the DRAM light response peaks in the near infrared region at 700 nm. The oyster meats appeared to be black on a white background and the projected area could be determined by simply counting the black pixels.

Because there was no detector mechanism to determine if oysters were in the camera field of view pictures were taken as fast as possible. Each picture was checked to determine if it contained a whole image of an oyster. Pictures were discarded if oysters were partly outside the field of view. The system was able to receive and analyze a picture with one oyster image in 176 ms which permits a grading speed of five oysters per second if the oysters are evenly spaced and distance between them and conveyor speed are synchronized to let an oyster appear in the field of view every 176 ms. Calibration of the vision system showed that conveyor speed had a considerable influence on the size of the images due to a system inherent fixed exposure time of 1/15 second. At slow speeds object images were significantly bigger than at high speeds. Therefore the system had to be calibrated for a fixed speed. Fifteen RPM were chosen. The calibration equation for the relationship between pixel and volume using equation [1] was found to be :

$$V = 1.7 + 0.01875 * P \qquad [3]$$

where:

V = Volume [cm^3]

P = Pixel

with an estimated measurement error of $\pm$ 3.0 cm^3 at a 95 % confidence level. The system was further developed to grade three oyster images per exposure.

4. Design of Oyster Meat Sorter

This chapter will describe the development of the sorting machine, the principle of operation of the finished developed machine, each machine component and software, and how the components fit together. Finally, a list of processing time for the image system and the sorting software is included because processing time directly influences the grading and sorting speed.

4.1 Influence of Previous Research on Design

Oyster meats are soft, gray-white, wet and slick objects consisting almost solely of muscles. They are easily pierced, cut, or torn which promotes their decay and should be prevented. After shucking they are usually stored in wash tanks before they are graded and packed. They need to be handled gently and should stay wet during the entire time between shucking and packing. These restrictions and the knowledge about correlated physical properties like weight, projected area and volume (Awa et al., 1988; Diehl et al., 1988) served as input to the design of the sorting machine. However, the design of the sorting machine described in this thesis was synthesized completely by the author.

It was intended to use the grading system described by Awa et al. (1988) (see Section 3.3). However, the integration of that particular system in a sorting device introduced so many problems, that it was abandoned despite of the fact that systems using backlight are easier to control. A discussion of these problems and their influence of the sorter design follows.

An assumption had to be made about the minimum space between oysters. Twelve to 15 cm between oysters was judged to be enough distance to operate an ejection device (mechanical wipers or water jets, preferable water jets because the chance of damage to the oyster meat mucous layers is minimized) on one oyster in the line without touching the neighboring oyster meats. Assuming a grading rate of 2 oyster/picture and 2.5 pictures/s results in a translational velocity between 0.3 m/s and 0.375 m/s. The translucent, rotating disc used by Awa et al. (1988) offered an ideal solution accommodating both grading and sorting equipment above and a light source protected from moisture below. Rotational motion exerts a centrifugal force F_c on the oyster meats dependent on the mass m of the oyster meats, distance r between the meat and the center of rotation, and angular velocity ω of the meat :

$$F_c = m\,\omega^2\,r \quad = m\,v^2/r \qquad [4]$$

The angular velocity is the quotient of the tangential velocity v and the radius r. The oyster mass and the angular velocity are given constants for each meat and the system, respectively. F_c is kept small by a large radius. The size of the radius is limited by practicality and costs, and these influenced the choice of 1.0 m for the disc diameter. A dry disc does not present any problems. Awa et al. ran his disc 17 RPM to grade 5 meats/s and no slippage of meats occurred because the disc was kept dry at all times. However, under sorting conditions, the conveyor would be wet from all the water drip that oysters carry out of the wash tank. Tests were run with oysters placed 40 cm apart from the rotational axis at the minimum required rotational speed of 7.2 and 9 RPM. The disc was misted with a plant sprayer, the meats were placed on it, and the speed was slowly increased. The oysters began to slip at approximately 9.0 RPM on wet plexiglas and even earlier on wet glass. This was considered unreliable. A sandblasted surface increases the friction coefficient to a sufficient level but the oyster meat skin could be damaged and skin particles in the grooves would make it difficult to keep the surface clean. The disc idea was discarded.

Wet oyster meats can be conveyed translational at 0.3 m/s up to 0.45 m/s on materials such as plexiglas, glass, PVC and polyurethane. Unfortunately, there is no flexible and translucent material available for a translational conveyor belt that doesn't scratch easily (scratches appear in images) or that can be used for conveyor belts of more than 4 m length. A slat conveyor consisting of plexiglas slats would also damage oyster skin. The light source underneath the belt or slats would have to be protected against humidity in a special way or the light source has to be located outside the conveyor area and the light would be reflected up through a mirror. This introduces the problem of uneven backlighting through water drops on the mirror which show up as dark spots in the picture.

Another possibility is a closed light box with a tilted surface. The oyster meats drop onto that surface and slide across the vision field onto a belt conveyor. The sliding speed is dependent on the coefficient of friction and oyster weight. The dynamic RAM vision system can only grade accurately if oysters move at one constant speed because the image size of an object decreases with increased speed. This is an added source of error which made that type of vision system less suitable.

The 5 cm vision field width of the camera used by Awa et al. is not wide enough to allow pictures of oysters in all orientations. An average oyster is 6 to 7 cm long and wouldn't fit into the field of view picture. A shorter focal length lens would provide a wider field of view at the same object-lens distance, but wide angle lenses introduce picture distortion as a source of error. The main problem was the camera's fixed exposure time which would make a mechanical shutter (rotating disc with holes) or strobed light necessary to make the whole system speed independent. A mechanical shutter would cut down light intensity below an unacceptable level. A strobe light would deliver enough light and make the system speed independent.

All those problems associated with a backlit, dynamic RAM vision system suggested to use a vision system with frontal object lighting and a translational conveyor.

4.2 Principle of Operation

The prototype oyster meat sorter described herein solves two tasks. It grades and sorts oyster meats. Connected to a singulation device at the front and to a container filling machine at the rear it would automate the entire fresh oyster meat packing process, excluding shucking.

Two lines of oyster meats are dropped from a singulation device or by hand onto a dark conveyor belt which moves them past a machine vision system. The vision system takes high contrast pictures of two meats in one picture, analyzes the images for projected area and grades the oysters into three different categories: *Standards*, *Selects* and *Counts*. In this research, *Counts* were considered to be oyster meats smaller than *Standards*, or *very small* according to the FDA standards of identity. In commercial practice *Counts* are meats larger than *Selects*. The limits between the classes are adjustable in software to any value. The conveyor moves the graded oysters past a row of high-impact water jets that spray the oyster meats off the belt into different containers beneath the belt. A personal computer and a input/output controller board control the machine vision system and the action of the jets. The computer sends information about size and position of graded oysters to the controller board which keeps track of all the graded oysters on the belt and turns the jets on and off at the appropriate time.

4.3 Conveying Equipment

A skidplate belt conveyor transports the oyster meats through the camera view field and later past four jet nozzles used to sort the oysters in large (*Selects*) and small (*Counts*) sizes while the remaining medium (*Standards*) sizes fall into a bucket at the end of the conveyor (Figure 5). The conveyor was designed to be as small as possible to fit into small rooms and be easy to assemble.

The endless belt is light-duty, darkgreen, 35.5 cm (14") wide, 0.7 mm (0.03") thick and 396 cm (13') long. Its material is FDA approved for contact with raw meats and consists of polyurethane. To enhance the contrast of the pictures the belt had to be as dark as possible and darkgreen was the darkest food approved belt color available. The belt has a 180 cm (71") transportation length between two crowned, 11 cm (4.3") diameter pulleys. The crown face is necessary for belt tracking and amounts to 4 mm (1/8") more diameter in the center of the pulley. The width of the pulleys are 5 cm (2") wider than the belt to give the belt an opportunity to track (Smith, 1972). Both pulleys are lagged with a 5 mm (3/16") thick rubber strip to increase their efficiency and protect their surfaces against corrosion. The 25 mm (1") pulley shafts are fixed to the pulleys with tapered keys to prevent sideways movement. The drive pulley shaft is mounted in two self-aligning flange ball bearings and the tension pulley ball bearings are mounted in two take-ups that allow a linear adjustment of 7.6 cm (3").

Conveying of approximately 10 to 15 oyster meats at one time imposes practically no

Figure 5 Oyster Meat Conveyor

load on the conveyor. This allowed a skid plate conveyor design without snub pulleys. The 16 gauge, stainless steel skidplate has two tasks. One is to provide a perfectly horizontal, straight belt surface to prevent unwanted sliding of oyster meats during transport, second the sides of the skid plates are bend down in a 45° angle to guide the meats into the containers beside the conveyor (Figure 1).

Skidplate, belt, and pulleys are mounted on a welded steel frame made from square tubing which is bolted onto a stand allowing the belt to be changed easily. The welded stand is 117 cm (46") high, 175 cm (69") long and 71 cm (28") wide and runs on casters for easy transportation. The stand width allows an assembled transportation from one room to another through doors of 120 cm minimum width. Drive motor and its controller, vision system, spray valves and their controls, and container shelves are mounted on the stand but are detachable.

The conveyor is driven by a speed-controllable 90 W gear motor (*4Z382,* Dayton mfg. Co.) with a gear ratio of 28:1 powered by a 90 VDC speed controller. The speed controller is designed for applications with constant and diminishing torques as required by conveyors allowing belt speeds from 0.05 m/s to 0.5 m/s (*4Z829,* Dayton MFG. Co.). The controller can be operated in FORWARD, REVERSE and BRAKE modes. It also has an internal "soft start" and a dynamic braking circuit stops the motor with no appreciable external inertia. Under experimental conditions with frequent stops, this is an advantage. A gearbelt transmission drives the conveyor with a gear ratio of 1:1. Gearbelt drives combine the flexibility of belt drives with the advantages of chain and gear drives. They eliminate the nuisance of slippage, which needs to be avoided in the closed loop oyster position control system. They can submit peripheral forces up to 5000 N without lubrication and corrosion of chains, which would be disadvantageous in wet environments. The power is transmitted by positive engagement of belt teeth and pulley grooves, as in chain drives, rather than by friction as in belt drives. Because of the slight side-thrust of gearbelts in motion, both pulleys are flange (Roloff et al., 1983).

Two drawings are included in Appendix A showing the construction and measures of the conveyor bed frame and the stand and its attachments. Both frames are welded together except where holes indicate bolt connections. All conveyor parts are listed in Appendix B under *Conveyor Components*. The list includes identification number used in drawings, number of parts used, a short technical description of each component, manufacturers catalog number and manufacturer.

4.4 Machine Vision System

The main components of the machine vision system - a monochrome television camera, a xenon flash light and an optical oyster detector - are housed in a box at the front end of the conveyor. The boxes lower, open end covers the belt width providing a 2 cm gap between belt surface and box. The oyster meats are conveyed underneath the shield of the box where their image is taken. The essential synchronization of the flash light and capturing the corresponding frame/field from the camera is achieved by a photoelectric switch. An infrared light emitting diode (IRED)/phototransistor switch is opened when an oyster arrives at a predetermined position in the field of view. The phototransistor signal is converted to a suitable signal for the Asynchronous Communications Adapter (ACA) of a personal computer (PC). The signal is detected by software which issues immediately a signal sequence to a digitizer/transmitter board to grab the next picture frame. When the sync-signal of the next frame scan is sent to the PC, the computer issues the trigger signal for the flash to light the scene. The analog TV camera signals are digitized and sent to the PC. The images are analyzed for projected area of the oyster meats. The software sends the grading results to the spray valve control board which takes care of the sorting. Figure 6 shows a schematic of arrangement and logical connection of the image acquisition equipment.

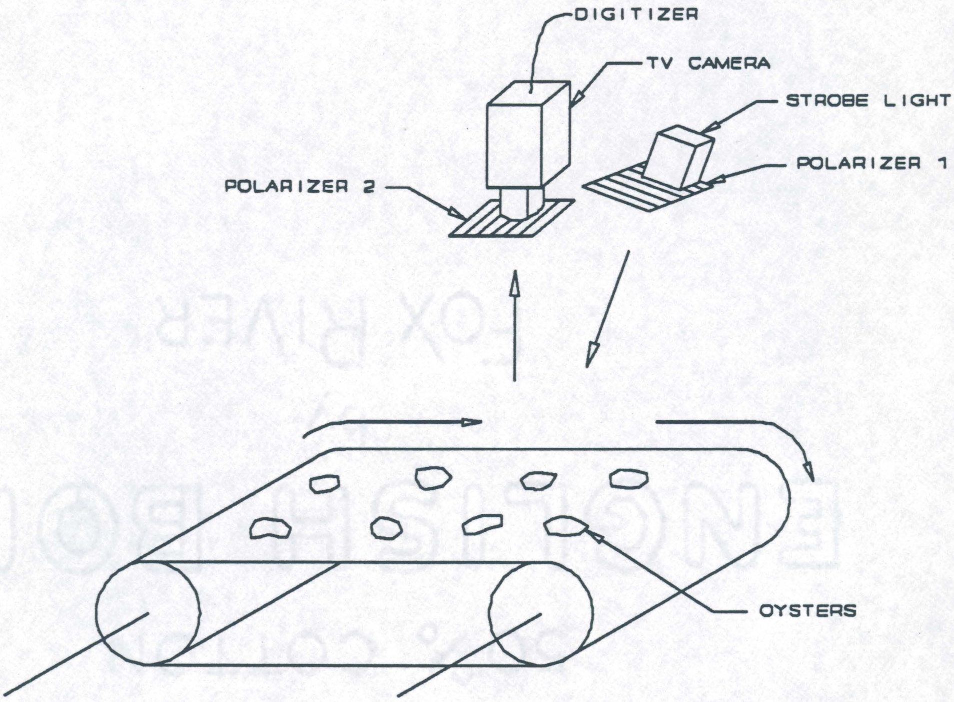


Figure 6 Vision System for Oyster Meat Grading

4.4.1 Image Acquisition Equipment

The image sensor is a *Panasonic WV-1410* monochrome television camera. The camera is a Electronic Industries Association RS-170 standard vidicon tube (Type 20PE13A) device. This camera was chosen because it has a relative high picture quality, high sensitivity, works satisfactorily with strobe lights, has an extended spectral response and is inexpensive (Batchelor et al., 1985). The disadvantage of non-uniformity of scan was not regarded important enough to outweigh the advantages for this application. The camera is equipped with a 16 mm C-mount lens, which has variable aperture and a focus adjustment. The necessary illuminance to create a picture is 5 lux with this lens.

Picture grabbing and synchronization of flashes with that picture is closely related to the way a TV camera operates. Such a camera employs a vacuum glass tube as the image sensor, having a photosensitive target which is placed in the image plane of the camera. The readout mechanism involves the use of an electron beam generated in the tube which scans the target to sense the charge pattern created by an image falling on the tube face (Figure 7). The electronic image is created by scanning the electron beam horizontally across the front of the camera tube in a back and forth motion until eventually a picture, or "frame" is completed. A frame is created by scanning from top to bottom twice (Figure 7). This is analogous to integrating two scans of a typewritten page. With one scan all the printed characters are captured and with the second scan all the line spaces are captured. Each of those scans is called a field and

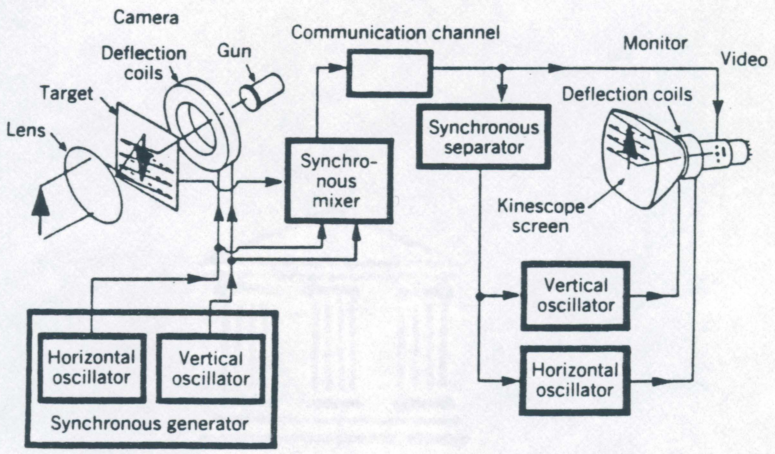


Figure 3.1 EIA RS-170 vidicon camera TV transmission.

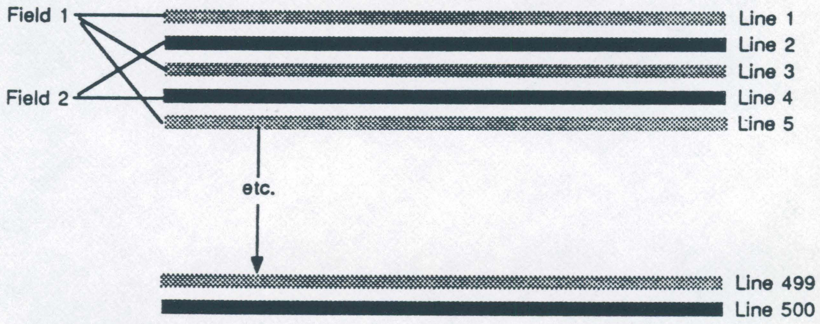


Figure 3.2 Interlaced scanning structure (courtesy of Visual Information Institute).

Figure 7 Principle of Operation for Vidicon Television Camera (Zuech, 1988) is one half of a frame; if both are interleaved or interlaced, a double-spaced letter page results. The camera sweeps from left to right as it captures the electronic signal and then it retraces when it reaches the end of a sweep. The retrace is triggered by a portion of the video signal called vertical sync. Scanning of a new line is triggered by a horizontal sync pulse. Systems used in the United States scan a new frame every 30th of a second and a field is scanned every 60th of a second. Each field has 263 lines and a completed frame has 526 (Batchelor et al., 1985).

The analog signal output of the TV camera needed to be transformed into digital information to make it accessible for computerized image processing. The digitizer used is the *ImageWise* frame grabber (Micromint, Inc.). The ImageWise frame grabber requires a standard video input source with 1 V Peak-to-Peak signal and a 75 Ω impedance that TV cameras, Video Cassette Recorders (VCR) and laser disc players supply. Its main component is a RCA 3306 6-bit flash A/D converter operating at a 5 MHz rate converting the analog picture into a 244 line by 256 column digital picture totaling 62,464 picture elements (pixel) with 64 gray levels (0 is black and 63 is white) in 63.5 μs real time (Ciarcia, 1987a).

Internal software stores the picture into a 64 kbyte buffer, compresses it (optional) and transmits it as RS-232 standard serial data to a computer. The RS-232 data are sent in 8 data bits, no parity, one start and one stop bit. The video data capture starts by detecting the vertical sync-pulse in a field and the digitized data are transmitted to the computer using the format shown in Table 2. The lower 6 bits contain the grayscale

Table 2 Structure of Data Byte Sent to and from Host Computer

Bit number								Bit definition
7	6	5	4	3	2	1	0	
TO host computer :								
0	0	x	x	x	x	x	x	Video data byte
0	1	0	0	0	0	0	0	Start of video field
0	1	0	0	0	0	0	1	Start of video line
0	1	0	0	0	0	1	0	End of video field data
0	1	1	x	x	x	x	x	Reserved
1	0	0	0	x	x	x	x	Repeat previous byte x times (0 to 16 reps)
1	0	0	1	x	x	x	x	Repeat previous byte 16x times (0 to 256 reps)
1	1	x	x	x	x	x	x	Reserved
FROM host computer :								
0	0	0	1	0	0	0	1	XON, starts/restarts transmission
0	0	0	1	0	0	1	1	XOFF, halts transmission
1	0	0	0	0	0	0	0	244 x 256 resolution (full)
1	0	0	0	0	0	0	1	122 x 128 resolution (quarter)
1	0	0	0	0	0	1	0	61 x 64 resolution (sixteenth)

information while the others indicate picture configurations. The software is able to send the pictures in three different resolutions, high (244 x 256 pixels), medium (122 x 128 pixels) and low (61 x 64 pixels). The highest bit resolution was chosen for this application. To save transmission time the digitizer/transmitter software can compress each line of data by representing repeated bytes by a value and a repetition count. This reduces the number of bytes for each oyster picture in average by 8 to 10 times. Transmission speed is selectable between 300 and 57,600 bits per second. The highest speed was chosen.

Some of the described features are set through DIP switches on the board itself the others have to be set in the image processing software of the host computer. Table 3 shows the DIP switch settings for the oyster grading application. To get an image the host computer software has to send a byte containing the desired resolution and then a XON signal to trigger the digitizer to capture a field (Table 2). Black and white contrast can be adjusted through two trim potentiometers, one for white and one for black level adjustment. They were adjusted to get the maximum amount of contrast between the dark belt and the light colored oyster meats.

It is not possible to achieve the goal of 400 ms/picture transmission and processing time with the *ImageWise* digitizer/transmitter because the picture data transmission and expanding time takes more than 2 seconds. Due to a delay in funding for this project a vision system available from earlier projects was used to show that accurate grading is possible with frontlighting. This board will be replaced with a parallel transmission

Table 3 ImageWise Digitizer/Transmitter DIP Switch Settings

Switch	Position	Set	Meaning
SW1, SW2 and SW3			Serial bit rate
SW1	-	ON	
SW2	-	ON	57,600 bits/sec transmission rate
SW3	-	ON	
SW4			Delay from first vertical sync
	OFF	OFF	16 lines
	ON	-	20 lines
SW5			Run length encoding
	OFF	-	disabled (no compression)
	ON	ON	enabled (compression)
SW6			Compress +/-1-count changes
	OFF	-	encode (less compression)
	ON	ON	ignore (more compression)
SW7	-	OFF	Not used
SW8	-	OFF	Not used

vision system board transferring pictures in 100 ms into computer memory.

An IRED/phototransistor switch to detect an oyster passing by was installed close to the surface of the belt. It needs to sense objects of more than 5 mm in height across the width of the belt that means approximately 27 cm. The long emitter/receiver distance required a powerful diode. The *1N6266* infrared emitting diode (Gallium-Arsenide) has a power dissipation of 170 mW and therefore sufficient light intensity to be sensed by the Fairchild *FPT110* phototransistor at the required distance. The diode and the transistor were matched through comparison of peak emission wavelength (940 nm) and peak response wavelength (800 nm), both in the infrared range. The TV camera does not "see" the infrared beam which is located in the upper part of the cameras field of view because it is insensitive to the wavelength used. The low voltage transistor signal for a closed switch (no object) and the high voltage signal for an open switch (object between IRED and transistor) was used as an input signal for a comparator circuit with a National Semiconductor Corporation *LM324* operational amplifier operated through a +/- 12 VDC power supply. Figure 8 shows the circuit and its components. The operational amplifier output switches between + 10.8 V for a closed switch (no object) and - 10.8 V for an open condition (object between source and sensor).

Those two voltages and the picture data are received and analyzed by a Jameco *JE1017* (Jameco Electronics, 1988) microcomputer which uses a 16 bit 80286 microprocessor having selectable processing speeds between 6 and 12 MHz. Only the highest

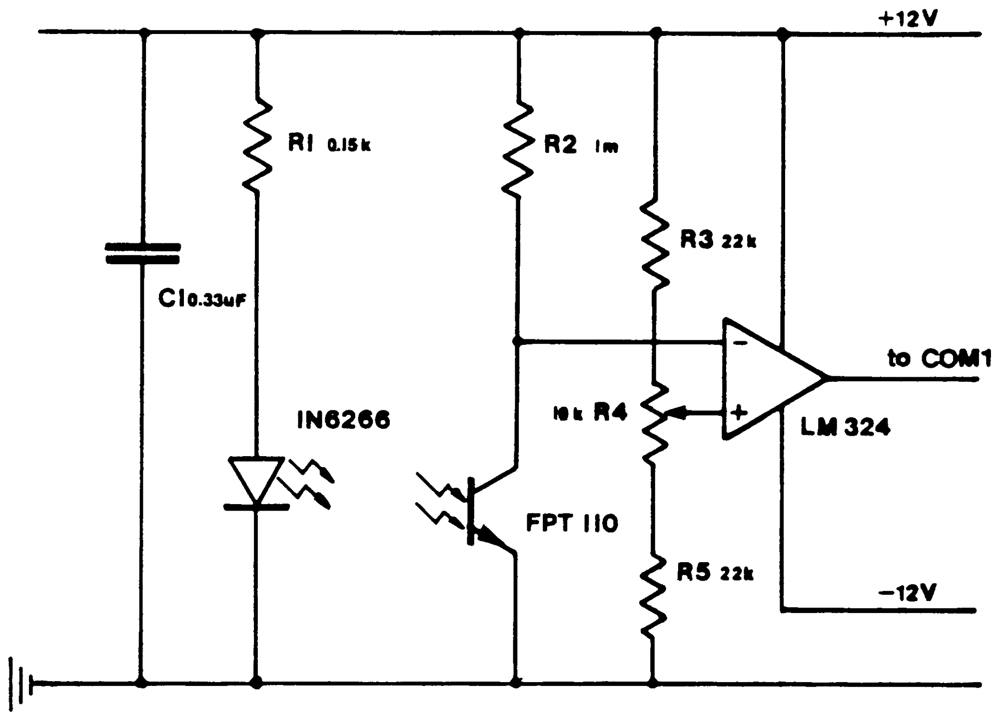


Figure 8 Comparator Circuit for Object Sensing on the Belt

processing speed enables the computer to receive data at a baudrate of 57,600 bits/second. At lower speeds, picture data were lost. A VGA graphics card allowed the display of digitized pictures on the screen for software development and adjustment of lighting and the camera. The comparator circuit signals and the picture data were received through Asynchronous communications Adapter 1 (ACA1 or COM1). The flash trigger signal is also issued through COM1. The communication line between the microcontroller board and the spray jet controller is established through ACA2 or COM2 using a baudrate of 9600 bits/second. A monochrome TV screen was connected to the camera displaying continuous video pictures to allow easy focussing and aperture adjustment of the camera.

4.4.2 Object Illumination

Two problems were encountered during lighting design for the vision system. First, specular objects, specifically water drops and the shiny surface of the belt, caused white spots and areas in the digitized picture. For oysters laying in a surrounding water puddle the problem became unsolvable. It was impossible to distinguish between water and oyster meats since both appeared to be white. This problem was solved using polarized light as described in the next paragraph. The other problem was related to the motion of the objects. The oyster meats move 5 mm during the capture of one field at conveyor operating speed of 30 cm/s causing a blurred image not suitable for area measurements. This problem was solved using strobed light also described later.

Specular reflection can be removed from the picture by polarizing the illumination light and then viewing the subject through a second, orthogonal positioned polarizer. The way light travels is closely related to how polarizers work. Light travels in form of transverse waves. In an ordinary light source the direction of the vibration of the light waves is random and is referred to as non-polarized. If the light vibrates in parallel waves it is called polarized and is often achieved through sheet polarizers. The polarizer transmits only light waves that are parallel to its axis absorbing 60% or more of the light falling onto the polarizer material. Polarized light hitting a specular surface stays polarized, while a truly diffuse surface depolarizes the light. A second polarizer in front of the camera lens, the axis crossed to the first one in front of the light source, passes half of the diffuse light and blocks all the specular light (Mersch, 1984). This method successfully removed all the glare and glint in the oyster pictures. The high light attenuation of two crossed polarizers introduces the problem of finding light sources with sufficient light intensity.

One source of very high light intensity is a xenon flash tube with a peak radiance between 800 and 1000 nm but still having 30% of its light output in the visible range. It has the widest spectrum of all discharge tubes used in machine vision and has been successfully used together with vidicon cameras (peak response wavelength 700 nm). Typical duration of flashes are between 10 μs and 1 ms. Flash tubes typically used in machine vision applications have power output up to 15 W at flash frequency of 60 Hz. Besides the flash tube a strobe light consists of a power supply with high voltage output, capacitors as energy storage devices which deliver high voltage at peak currents and a trigger transformer to initialize the discharge process (Gavenda, 1989). Because a strobe light designed for machine vision was unavailable for this project a photographic flash light (*AUTO 360PX*, Minolta Camera Co.) was used. At full output mode it provided a maximum flash range of 25 m for ASA 100 film speed at f1.4. The power output level could be varied in 9 steps from full power (Guide number 36) to 1/16 of the full power (Guide number 9). At low power output of 1/8 of full power it was able to deliver 3.5 flashes/s over a short time. The light intensity at 1/16 of full power is 130 lux/flash, which is sufficient to light the scene appropriately. A piece of sheet polarizer is taped in front of the tube and a polarizing filter is attached to the camera lens axis crossed as shown in Figure 2.

A box contains the vision system and protects the camera field of view from undesirable specular reflections from ambient light. The box interior was painted black to absorb the light not falling on the field of view. It also protects people in the work environment around the machine from disturbing flashes. The camera is mounted in the middle of the box cross-section. The camera lens object distance can be adjusted between 45 and 65 cm and the light source object distance can be varied between 45 and 70 cm with an adjustable angle of light incidence from the lens object axis between 8.5° and 30°. Those parameters where determined in preliminary tests of the image equipment.

4.4.3 Image Processing Software

Image acquisition, image analysis and communication with the sorting system was enabled through software written in TURBO PASCAL™ 3.0 (Borland, Inc.). The *ImageWise* frame grabber was supplied with picture receiver routines and image processing routines written by Ciarcia's Circuit Cellar in TURBO PASCAL™ 3.0 which were extensively used to develop the control software for this specific task. The developed software uses the frame grabber preset as described in section 4.3.2.

Three different software packages were developed. First, a graphics program displaying the grabbed pictures on a computer screen, allowing to view image manipulation to be viewed to determine the best image processing sequence. It was also used extensively to adjust lighting, position of objects, aperture, and the orientation of the polarizers. Second, a calibration program was developed that determines correction factors for a system inherent shift of brightness described in Section 5.1. It is also used to find the pixel/area and pixel/weight equation. Both programs use the same image acquisition and analysis routines as the actual sorting program uses which is described next.

The control program for the sorting process has three, consecutive tasks to accomplish. Initially it has to grab a picture frame upon object arrival in the field of view, synchronize the flash to that particular frame to light the scene, and receive the digitized picture. Then it has to analyze the picture, that is it has to "find" the oyster meat images in the picture and calculate their projected area in pixels, and estimate volume through the known relationship between projected area and volume. Finally, the software has to send that information to the microcontroller board to be processed for spray nozzle control (see section 4.4.3). All this has to be done in 400 ms to achieve the design goal. Time requirements for frame grabbing and transmission were fixed through the *ImageWise* system and its software. An image analysis algorithm that determines the area of the two oysters in the picture as fast as possible and that uses as little time as possible for information transfer between main computer and the microcontroller board had to be developed. A detailed description of time requirements for the different portions of software is given in the next section and a listing of the software source code is given in Appendix C.

Figure 9 shows a flow chart of the task sequence for the sorting program. After the program starts it sends information about the length of delays before turning on the nozzles to the microcontroller board (see Section 4.4.3). Then the program checks the amplified phototransistor signal constantly until the infrared beam received by the phototransistor is broken by the arrival of two oyster meats at a certain, adjustable position in the field of view. The signal is received through the Modem Status Register (MSR) of COM1 (address hex 3FE). Bit 6 of that address becomes 0 when objects are detected. The frame grabber gets a message sent through COM1 (address hex 3F8) to digitize the next frame. At the moment the digitizer sends the message through the Line Status Register (LSR) of COM1 that the next frame is ready to be taken, the flash is initialized. A high bit 0 closes the flash trigger circuit described in Section 4.3.2. The digitized picture data are sent in compressed format to the host

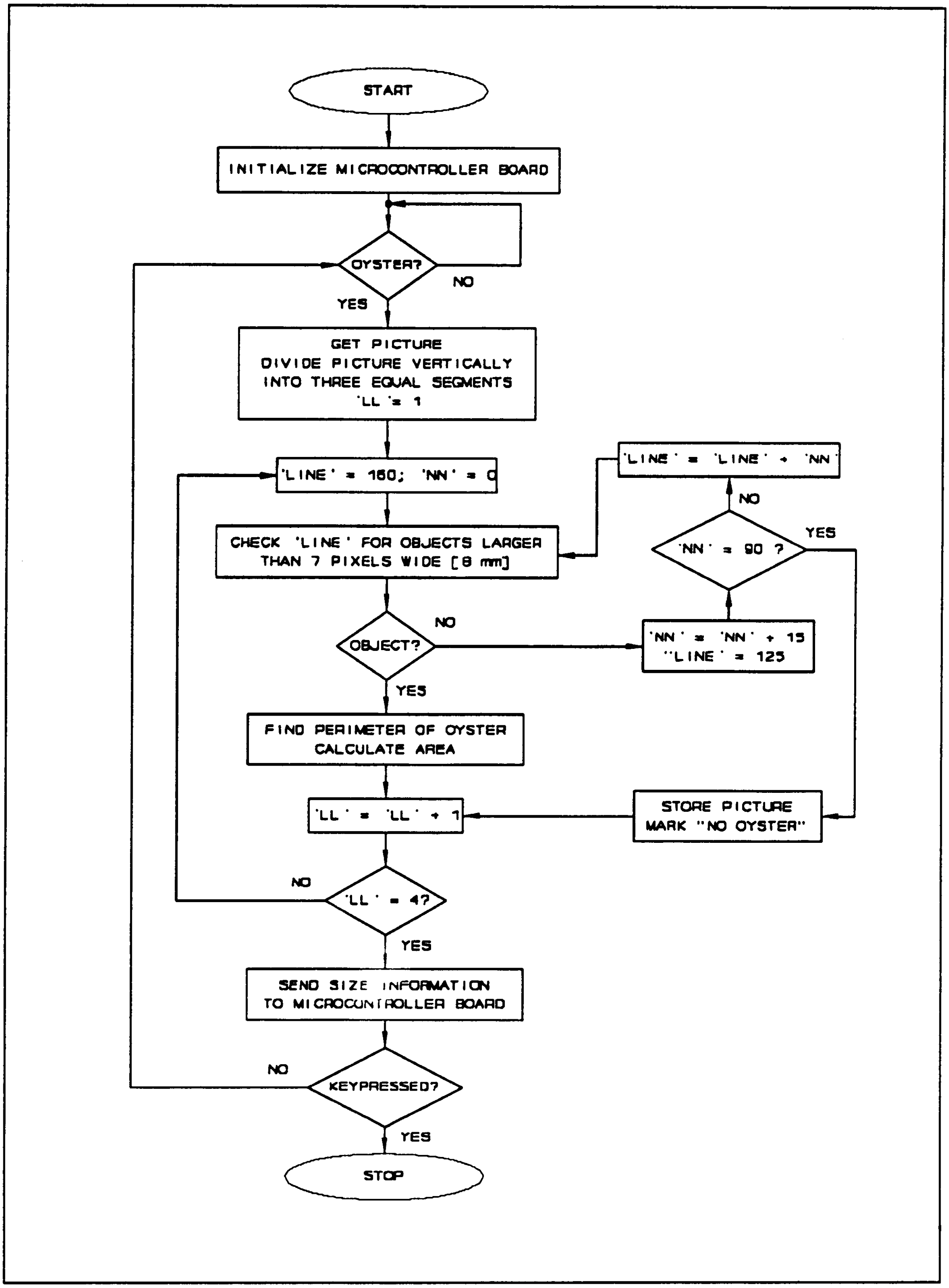


Figure 9 Flowchart for Grading Program

computer where it is expanded. After expansion the picture is segmented vertically into three sections, the left and the right section each containing an oyster image. The middle section contains a reference target for calibration purposes. The camera field of view and the oyster detector are adjusted to make oysters appear at line 160 in the picture. This line is searched first to find the objects. If there is no object in any one of the sections, line 125, 140, 155, 170 and 185 are checked for objects. No objects are marked with "No Oyster" on the screen during program execution and marked with a 'N' in the sorting result file stored on disc. Each object's area is calculated as it is encountered. The calculated areas in pixels are converted into actual area or weight using the equations determined in Section 5.4 and 5.5. The results are compared with the size criteria (Section 5.5) and graded. The size information consisting of a spray nozzle identification number between "0" and "4" is sent to the micro controller board for further processing.

The sorting program does not require even spacing between oysters. However, it requires separated oysters. The oyster meats should drop on the belt simultaneously so that they appear in the picture centered around the same line (160 in this case), because the sorting system operates with fixed distances between the position of the oyster meat and each nozzle when its image is taken. The spray duration is as long as possible (11 cm of belt travel) allowing the meats to be $\pm$ 3 cm off their assumed position (see Section 4.5).

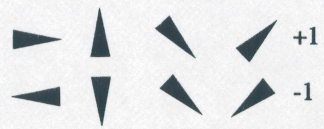
The area calculation subroutine (Appendix C) calculates the area like a planimeter.

The algorithm uses a fixed threshold and treats every number below that threshold as a 0 and every value above or equal as a 1, essentially creating a binary picture without changing any values. Figure 10 contains a small binary image to demonstrate how the area is calculated. The image has diagonally connected image pieces and several bays as an intricate oyster image might have. The algorithm searches along a predetermined line to find the right edge of the image (Line 5, encircled 1) and than traces the outline of the image following the arrows until it hits the starting point again. It multiplies each column number containing a traced 1 that has a 0 on its left or right side, but not on both sides, by +1 if the newly traced 1 is detected from a 1 on the left or below (single underlined 1's). If the trace line (of a 1 with only one 0 as a neighbor) comes from a 1 on the left or above, the column number subtracted by 1 is multiplied by -1. The columns are summed up as the algorithm traces along. Each 1 encountered with 0's left and right (double underlined 1) adds a 1 to the total sum. The algorithm is mathematically correct for images with no eight-connected image part appears as in the upper right corner of the image. Each double tracing between the same 1's causes 2 non-existent pixels to be counted. Eight-connected image parts could have been excluded from tracing but the error introduced was considered small compared to loss of valid image parts.

Row	1	2	3	4	5	6	7	8	9	10	11	12	13	Sum of Columns	Sum Pixel
1	0	0	0	0	0	0	0	0	0	0	0	0	0		
2	0	0	1	0	0	1	1	1	0	0	1	1	0	+1 -5 +8 -10 +12	= 6
3	0	1	1	0	1	1	1	0	0	0	1	1	1	-1 +3 -4 +7 -10 +11 +13	= 19
4	1	1	1	1	1	1	1	1	1	1	0	0	0	-0 +10 -9	= 1
5	0	0	1	1	1	1	1	1	1	1	1	0	0	-2 +11	= 9
6	0	1	1	0	1	1	1	1	1	1	1	1	1	-1 +3 -4 +13	= 11
7	0	1	1	0	1	1	1	1	1	1	0	1	0	+1 -1 +3 -4 +10	= 9
8	0	0	0	0	0	1	1	1	0	0	0	0	0	-5 +8	= 3
9	0	0	0	0	0	0	1	0	0	0	0	0	0	+1	= 1

Calculated Sum of Pixels :	59
Actual Sum of Pixels :	57

Conditions for Column Number Coefficient
Depending on Edge Direction :



Condition to Add a Single Pixel +1 :

0 1 0 Sequence in a Row

Figure 10 Edge Detection and Object Area Calculation in a Binary Image

4.5 Sorting Mechanism for Oyster Meats

The sorting system "knows" the position of all oyster meats on the conveyor after their image is recorded. The meats move at a constant belt speed while the system software classifies each before reaching the first spray valve. A signal input/output board interprets incoming signals about oyster positions and size, synchronizing jet on/off-time to spray each oyster into the appropriate container. Sorting system equipment is shown in Figure 11. Two jets on each side spray the meats into containers located on the corresponding side of the conveyor. The first jet on each side sprays very small (*Counts*) oyster meats off the belt. The second jet on each side sprays off oyster meats of the *Selects* category, while the more numerous *Standards* are conveyed off the end of the belt into a single container. The mechanical and electronical configuration are described in the following sections.

4.5.1 Mechanical Components

Water line pressures of 400 to 500 kPa, and flow rates of 16 to 18 l/min (as provided by residential water supply nets) are sufficient to run the sorting system. A grading rate of 2.5 meats/s at each side of the belt required fast acting jet valves. The *AA2AUH* jets chosen (Spraying Systems, Co.) were automatic air-actuated spray valves that cycle up to 3 times/s, with a maximum flow rate of 7.6 l/min. They operate at water line pressures of up to 4,000 kPa and require a minimum air pressure of 300 kPa

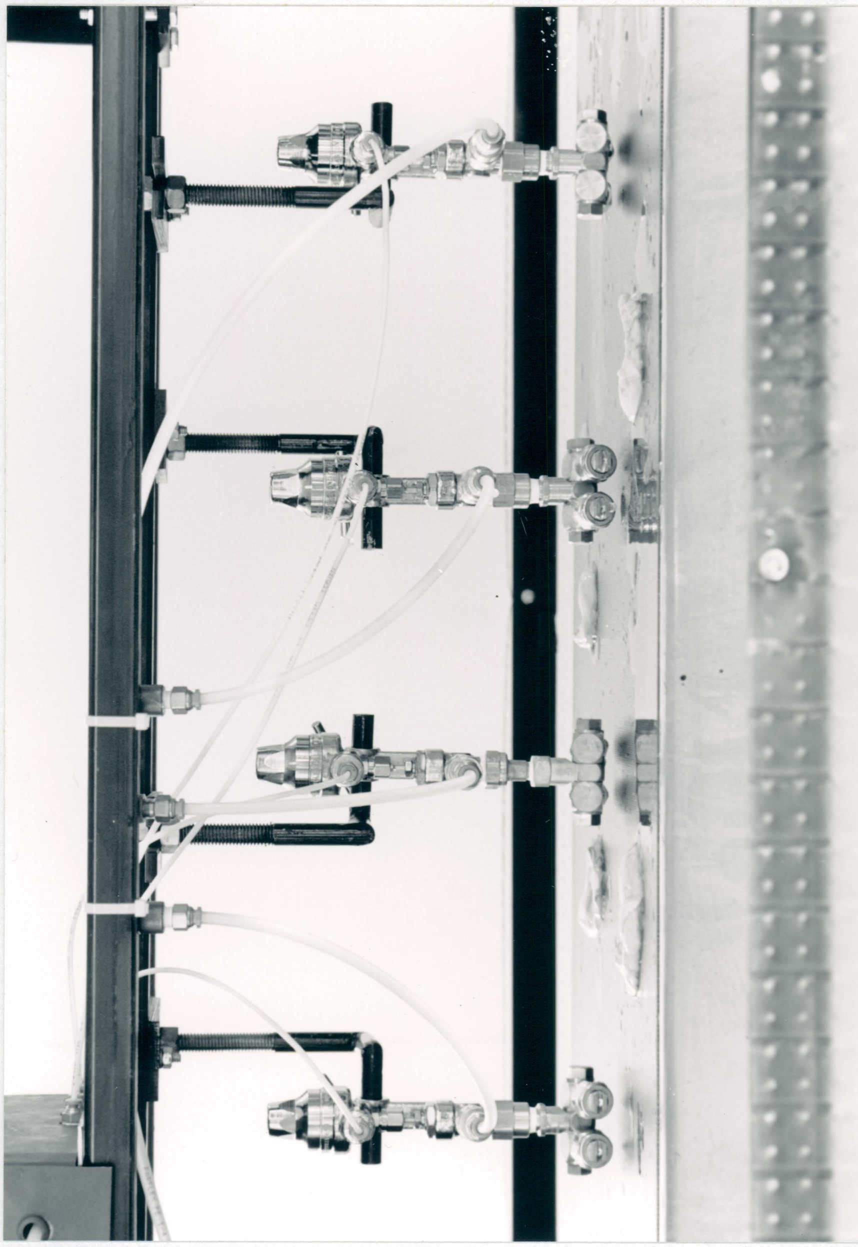


Figure 11 Configuration of Oyster Meat Sorting System

for actuation. Spray stream impact had to be high enough to move the oyster meats off the belt edge, where they slide down the skidplate into containers. Spray stream impact/cm^2 is dependent upon flow rate, spraying pressure, spray pattern (flat fan, hollow or full cone), and spray angle. A flat fan spray pattern has the highest impact. A narrow spray angle provides higher impact than a wider angle. Preliminary tests with various nozzles showed that two 25° spray angle units (*Unijet TP2502*, Spraying Systems, Co.) connected to a double swivel head with a flow rate of 1 l/min at 330 kPa operating pressure were capable of smoothly spraying oyster meats of any available size off the belt (Figure 12). The overall flow rate of 8 l/min did not exceed the flow rate available through a 19 mm water pipe (17.6 l/min). Water and air pressure are reduced through pressure regulator valves (RG). Air pressure operating the spray valves is switched on and off by four *M31EI* manifold valves (Humphrey Products). The 3-way, normally closed, fast-acting air valves are operated by *S505-OSJ410* solid-state relays (Continental Industries, Inc.) switching the 120 VAC required by the air valves. The relays require a 5 VDC input to switch which is provided by the input/output board. Electrical and air circuits are shown in Figure 13.

Belt movement (and hence oyster meat position) is monitored through an inductive proximity switch (*651810*, Veeder-Root, Inc.), sensing bolt heads on a disc connected to the return pulley shaft. The switch senses any metal targets moving laterally across its face if larger than the face width and closer than 5 mm to its face. The switch is normally open and operates on 12 VDC. Output signal are +12 VDC for an open switch and +1.1 VDC for a closed switch. These voltages are transformed through an

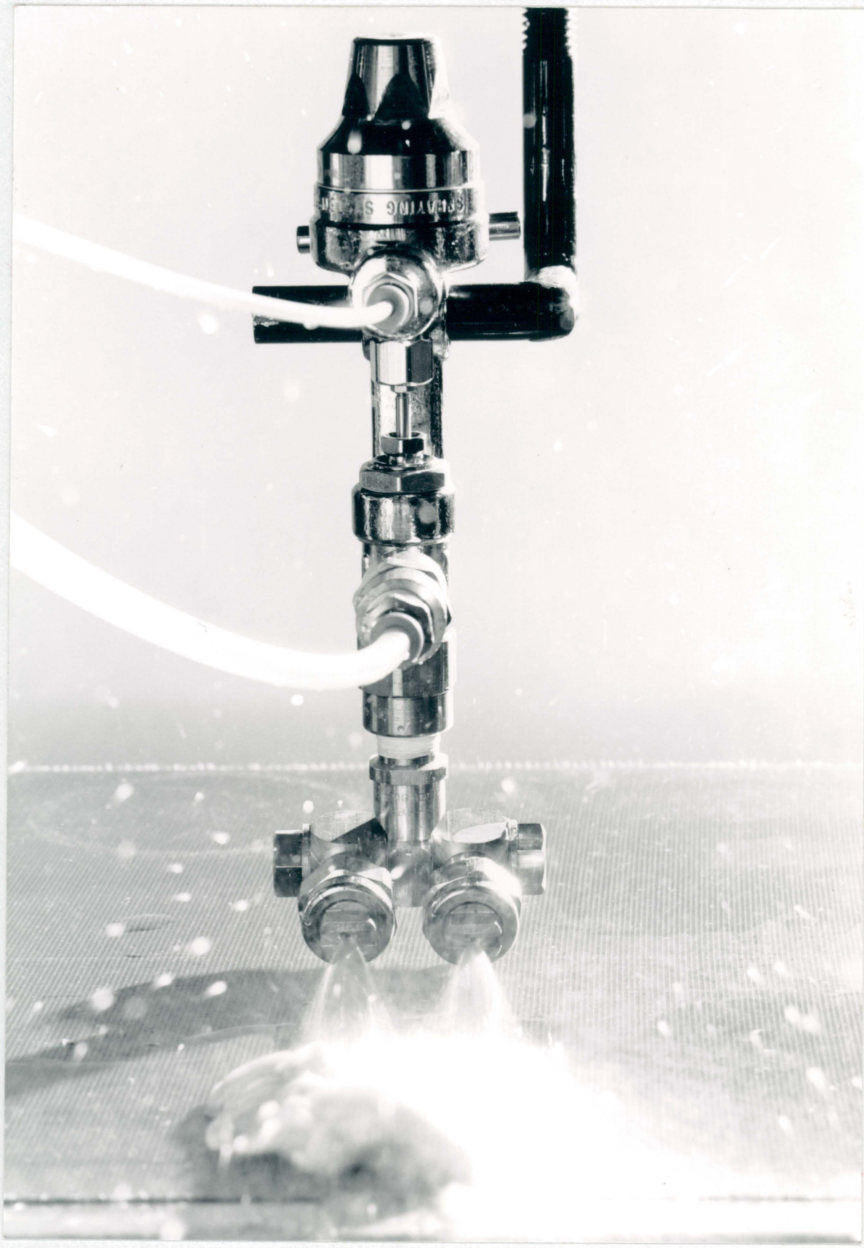


Figure 12 Spray Valve Configuration

SR1, SR2, SR3, SR4	solid state relays number 1 through 4
SV1, SV2, SV3, SV4	air valves number 1 through 4
JET1, JET2, JET4, JET4	spray valves number 1 through 4
R	resistor
RG	air pressure regulator

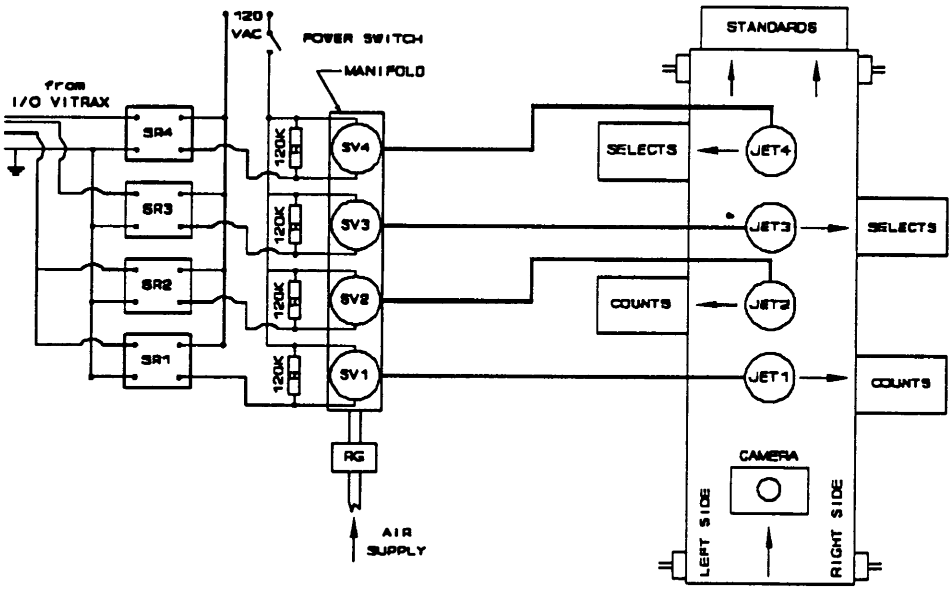


Figure 13 Connection of Air and Spray Valves

opto-isolator into +5 VDC and ground, which are used as input signals for the input/output board. Fifteen bolt heads evenly spaced on a 19 cm diameter plate indicate a 2.2 cm movement of the belt between switch pulses (55 ms between pulse occurrence at 0.4 m/s maximum belt speed). Figure 14 shows the disc and the switch configuration.

4.5.2 Electronic Hardware

All the different signals necessary to coordinate the sorting process are received and issued by the *VITRAX II* microcontroller board (Sintec Co.). The microcontroller board operates as a stand-alone computer system for dedicated digital/analog control. Its 64180 microprocessor executes EPROM stored, compiled software comprising an 8 kbyte MODULA 2 code for this application. The board needs a +5 VDC power supply. All its control signals are +5 VDC and ground. One of the boards two RS232 serial ports is used as a communication link with the PC, receiving information about nozzle distances, nozzle on-time, and oyster meat sizes at a baudrate of 9600 bits/s. Eight of the 24 input/output lines are used to control the spray jets. One of the two available counter/timers counts the microprocessor clock pulses between proximity switch pulses used to synchronize the action of the spray nozzles.

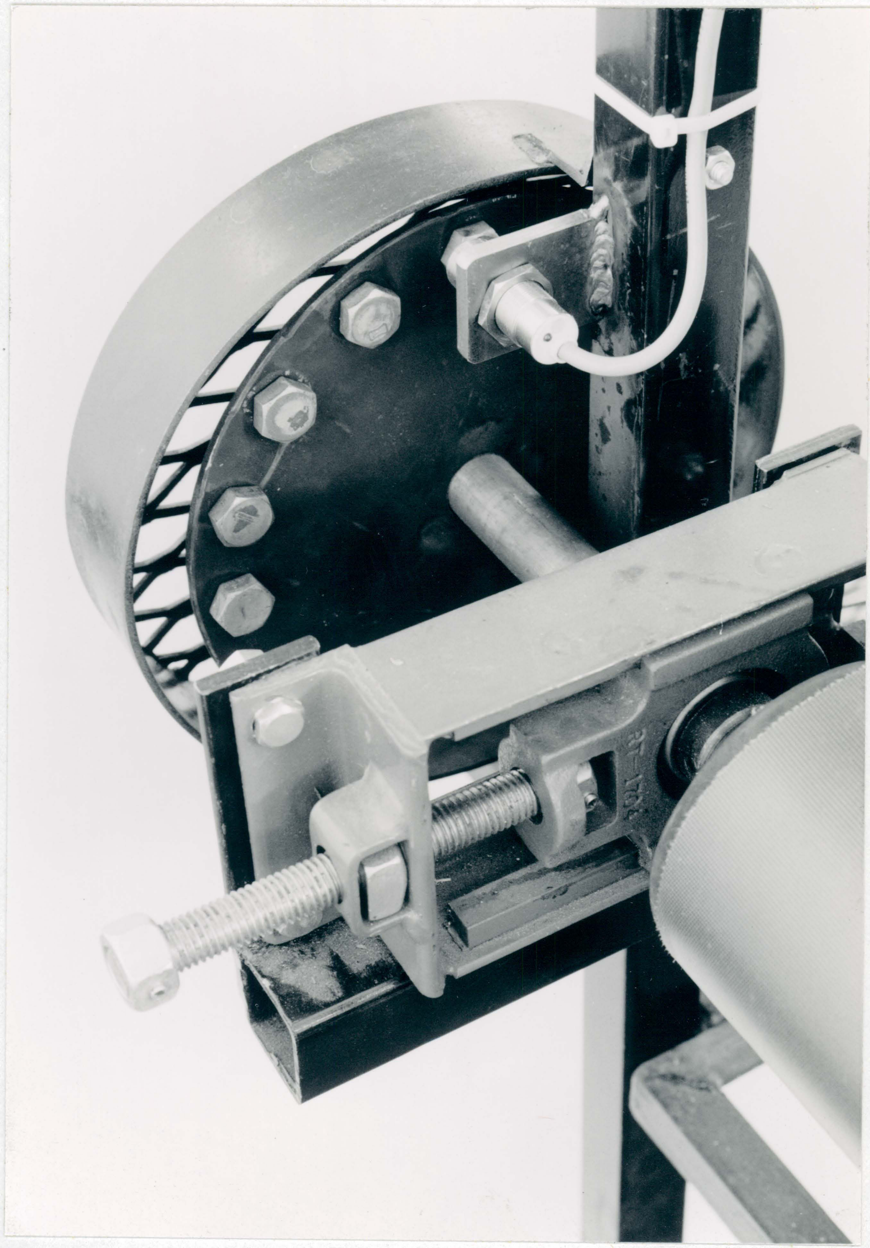


Figure 14 Belt Movement Monitoring through Inductive Proximity Switch

4.5.3 Software

Once the *VITRAX II* microcontroller board is powered up and the communication link between COM2 of the main computer and the board's RS232 port is established, the sorting control program starts. Figure 15 shows a flowchart that describes the sorter program control functions and their logical order. The program goes through a repetitive loop after the internal setup, terminated only by a power turn-off. The program goes through five small loops, checking different devices (flash signal high/low, serial port of PC, and belt position through inductive proximity switch) in subsequent order during the embracing loop.

The serial port buffer is first checked for a received character. A carriage-return character in the buffer indicates that a complete command sequence has been received and initializes the execution of that command. The control program requires five different numbers about pulse counts from the main computer before a sorting process is started, because it operates with predetermined distances between the point where the oyster meat image is taken and the four spray jets. The belt moves 2.2 cm between two pulse occurrences, which is not a sufficient resolution for precise nozzle operation. The resolution is mathematically increased to 0.22 cm by dividing the time between each occurrence by 10, calculating a smoothed belt speed and then triggering the pulse counter 10 times evenly between each pulse occurrence. The number of pulses between the moment the image is taken (flash signal) and the time the nozzles need to be operated are determined experimentally. Pulse count until operation of the first

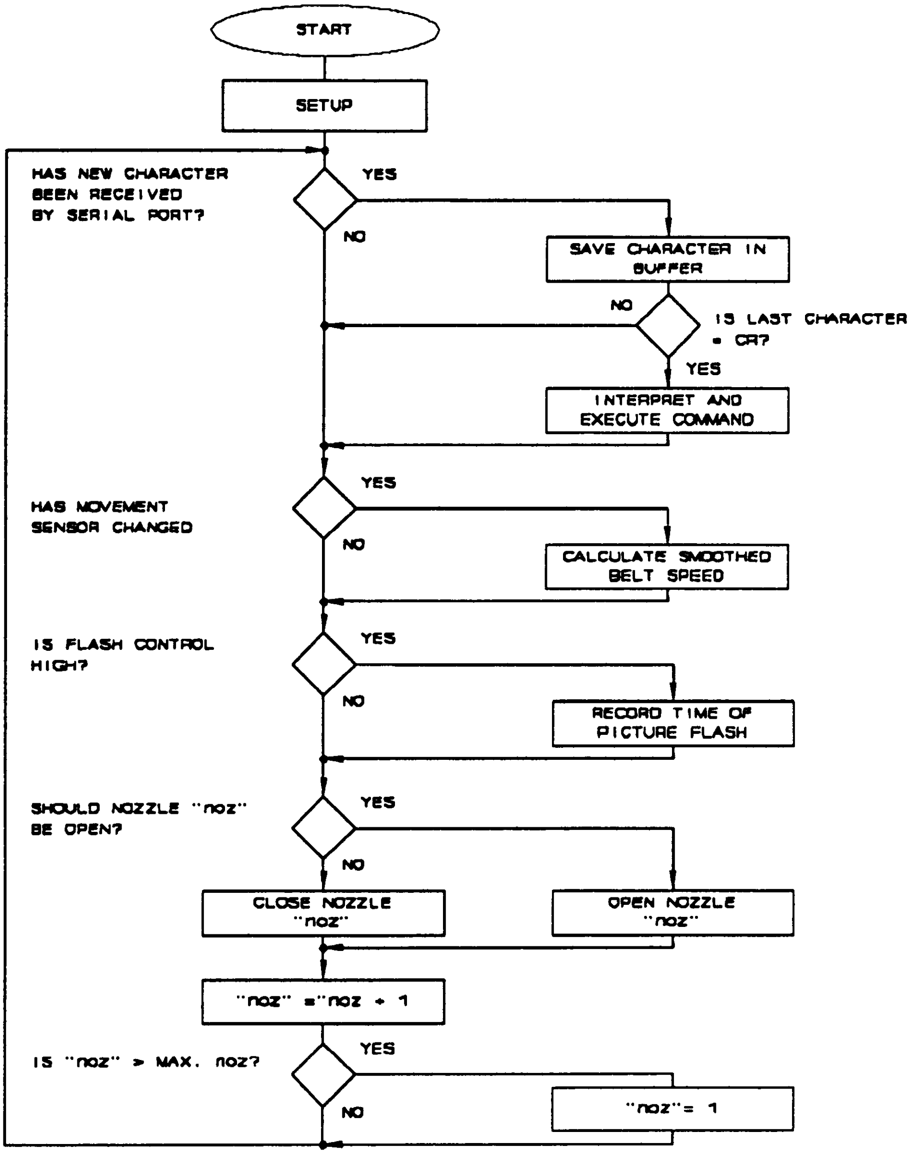


Figure 15 Flowchart for Spray Jet Control Program

spray jet (JET1) is 180 (39.6 cm), 255 (56.1 cm) for the second jet (JET2), 320 (70.4 cm)for the third (JET3), and 400 pulses (88 cm) for the fourth spray jet (JET4). Once actuated each spray jet remains open for 40 pulses (8.8 cm). The distance between the photosensitive switch and each nozzle are sent from the PC as three-digit numbers designated with the letter 'T' and a carriage-return (CR) character for identification. The nozzle-on time is sent as a two-digit number with an 'L' and a CR at the end. All set up information is sent to the board each time the PC sorting program is started. The only other information sent from the PC program is a two-digit spray jet identification number ended with 'N' and a CR character after each grading cycle. The first digit is either a '1' (*Count*) or a '3' (*Select*) assigned to the right side spray jets. The second digit is either '2' (*Count*) or '4' (*Select*) designating the left side spray jets. A '0' can appear at either or both places in the two-digit information, resulting in no spray action against meats destined for the *Standard* container at the end of the belt.

The flash signal triggers the pulse counter reading for each oyster meat pair arriving at the photosensitive switch. A fixed distance assumption between oyster meats at the imaging point and spray jets requires that the oyster meats arrive simultaneously at the switch position to be sprayed properly off the belt.

The program's main loop must be cycled fast enough not to miss a signal. Each time the loop is run through every signal is checked once and the software never waits for a signal. Each time it runs through it turns only one nozzle on or off. As the loop is cycled in 2.8 ms, it cannot miss the two critical signals: the flash signal and the inductive proximity switch signal. The flash signal stays high for the entire time of picture transmission and the pulse signal occurs every 55 ms at highest operation speed (0.45 m/s). The meats move 1.1 mm at highest speed during one loop cycle, which is insignificant.

4.6 Processing Time

One grading cycle time period requires approximately 3.2 s, which does not meet the design goal of 2.5 oyster meats/s on one side of the belt. The long cycle time accounts for an average oyster picture with approximately 18 kilobytes in compressed form. Nearly 97% of the cycle time is necessary to transmit (2.5 s) and expand (0.4) the picture. In the remaining 0.14 s, oyster meats are detected, their areas are calculated, then classified into one of the three size categories, with the resulting information sent to the input/output board. These time measurements are the average of 10 repetitiously transmitted and expanded pictures of two large oyster meats, including the metal reference target and an average of 100 picture analysis cycles for large oyster meats including area calculation of the reference target between the two oyster meats.

It is possible to replace the systems serial vision hardware with parallel transmitting image processing boards, which can transfer 62 to 256 kbyte into computer memory in 0.1 s. The result would be a grading cycle of 0.24 s for two oyster meats. The overall grading rate would than be 8.2 oyster meats/s for the machine, which is more than originally desired.

The sorting software operating the microcontroller board cycles in 2.8 ms, theoretically allowing 8.5 m/s belt speed and certainly not limiting the entire system.

The time evaluation for the computer controlled components shows that the maximal belt speed of 0.45 m/s (no slippage of oyster meats up to this speed) is the limiting factor for the maximal possible sorting rate for the machine.

5. Calibration of Machine Vision System

It was observed that the number of pixels counted varies from one image to another, imaging one object in the same place under unchanged lighting conditions. That variation could be a result of light intensity differences from flash to flash, fluctuating sensor response, uneven light-sensitive coating on the sensor surface, or any combination of the three. Investigations of the magnitude of those variations and their influence on measurement of projected object area are described in this chapter. The method of adjusting lighting conditions to avoid recalibration for each sorting run is also described. Oyster meat images appear in two locations on the sensor surface each of which may have different light intensity response. Several tests were performed to determine if there is a significant difference in number of pixels measured. One test was conducted to determine if the use of electronic flash really "freezes" the images of objects moving past it determining if object speed has any influence on pixel count. After the description of those preliminary tests, area measurement calibration is described. Gray-painted, rectangular metal targets were used for all tests. They have the advantages of uniform appearance, sharply defined edges, and their area is easily measurable. Awa (1988) described a calibration procedure for oyster meats measuring the projected area by using the vision system and an independent method. He took a photographic picture of each oyster meat with a grid of known areas behind it before taking a corresponding digital image. The oyster meat area was determined from the photographic print with a digital planimeter. It was decided to avoid that time consuming and elaborate procedure using the high correlation between weight and volume. Final calibration before sorting oyster meats was performed determining the relationship between their weight and the number of pixels that comprise their images. Statistical analysis of all tests were done through the Statistical Analysis System (SAS) software package (SAS Institute, Cary). Collected data are listed in Appendix D.

Tests on the vision system were performed under operational conditions. Oyster meats or other objects were conveyed through the field of view on both sides of the conveyor belt, with a parallel distance of 14 to 22 cm between them. Objects on the left or right side of the belt (from the direction of the camera toward the spray jets) appear on the upper left side of the digital pictures and will be referred to as on "LEFT side of the belt" or "RIGHT side of the belt" throughout the following chapters. A field of view covering the width of the belt, obtained at a object-lens distance of 54.5 cm, was more than sufficient to capture images of passing oyster meats. The field of view at this distance was 30.5 cm wide and 22.4 cm long, with a resolution of 1.09 mm^2/pixel. The electronic flash was adjusted at a 12.5° angle to the lens-object axis, at 68.5 cm distance to the conveyor belt. The flash intensity was set to 130 lux received at the object. A lens aperture of f/5.6 was found to be the optimal aperture when used with two perpendicularly crossed polarizers. The lens was focused on various objects under bright ambient light without polarizers while observing the picture on a video screen before any digital picture was taken. Subsequently the polarizers were variably crossed observing on the video screen the disappearance of glare. Rough adjustment of the aperture was performed by visual observation of the video screen during flash exposure. Final adjustment was accomplished by viewing histograms and graphic displays of digitized pictures. It was observed that the pictures containing gray metal targets could be adjusted to exhibit very high contrast through the black and white level trim potentiometers on the digitizer/transmitter board. The potentiometers were adjusted until the picture histogram showed an extreme bimodal distribution as shown in Figure 16. Pictures containing gray targets consist predominantly of black (gray level 0) and white (gray level 63) pixel. Pictures with this particular gray level distribution were attained by adjusting the incoming video signal below 0.90 V as black, and above 0.95 V as white. This set-up was maintained throughout all tests performed with metal targets.

5.1 Repeatability of Pixel Readings and Lighting Conditions

For industrial application the vision system should be readjustable without calibrating for each use. This was achieved by use of a gray rectangular metal reference target (731 mm^2) placed on a small platform 2 cm above the middle of the conveyor belt.

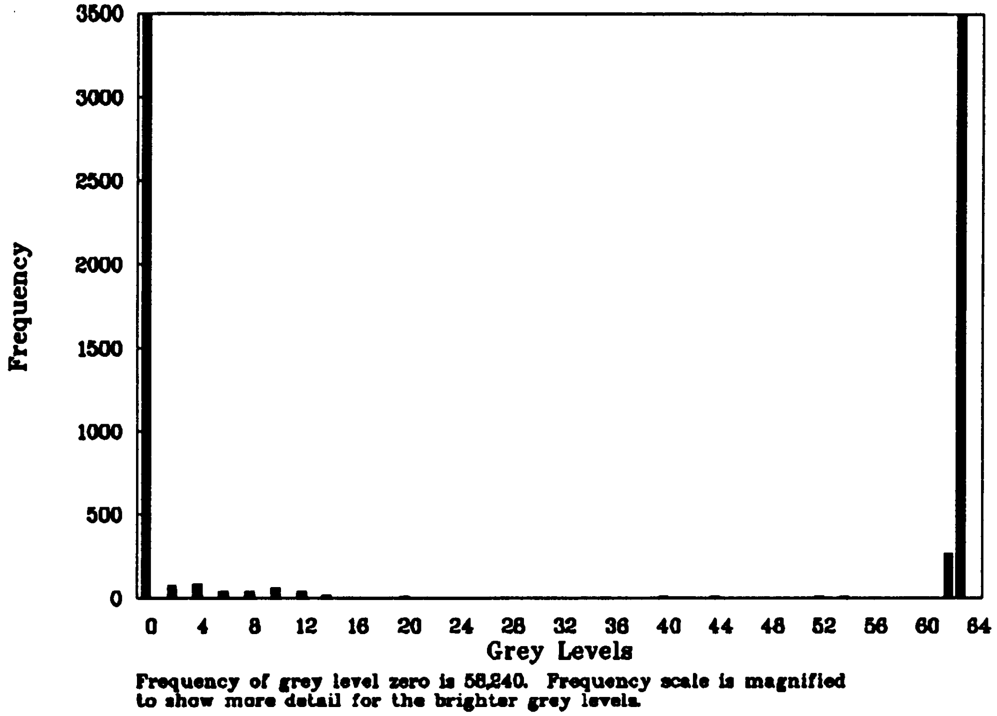


Figure 16 Gray Level Distribution for Pictures Containing Metal Targets

The platform keeps the target in place while pictures of moving objects are taken. The platform is covered with belt material to not disturb uniform background. The targets pixel area is calculated every time a picture of other objects is taken.

It was observed that a mean of 30 pixel readings was stable within $\pm$ 10 pixels for the reference target. Light and lens aperture could be adjusted to achieve a mean of 730 to 800 pixels without obvious deterioration or overexposure of the picture. The mean of 30 pixel readings for the reference target was adjusted to 776.1 pixels (Appendix D. *Uncorrected Pixel Data*). Frequency distributions of pixel readings for 30 pictures with targets remaining in the same position exhibited a strongly defined bimodal distribution. Figure 17 shows an example of a bimodal distribution for readings of a 747 mm^2 target referenced under "original pixel". F- and t-tests, which were intended to be used for statistical analysis, require normally distributed observations to be valid. An equation was needed to transform the bimodal distribution into a normal distribution. Investigating data with two other gray targets in the picture, it was observed that more pixels were counted for the other targets if the number of pixels increased for the reference target. Several Pearson correlation coefficient analyses showed very high correlation coefficients (99 percentile). For the data shown in Appendix D (*Uncorrected Pixel Data*) the correlation coefficient between the reference and the left target is 0.986, between the reference and right target is 0.99. Afterward it was noticed that all three target locations show repeatable different percentage standard deviations, the highest for the reference target (approx. 3.5 %) and lower for the other targets (approx. 2.3 % for the left and 2.8 % for the right target).

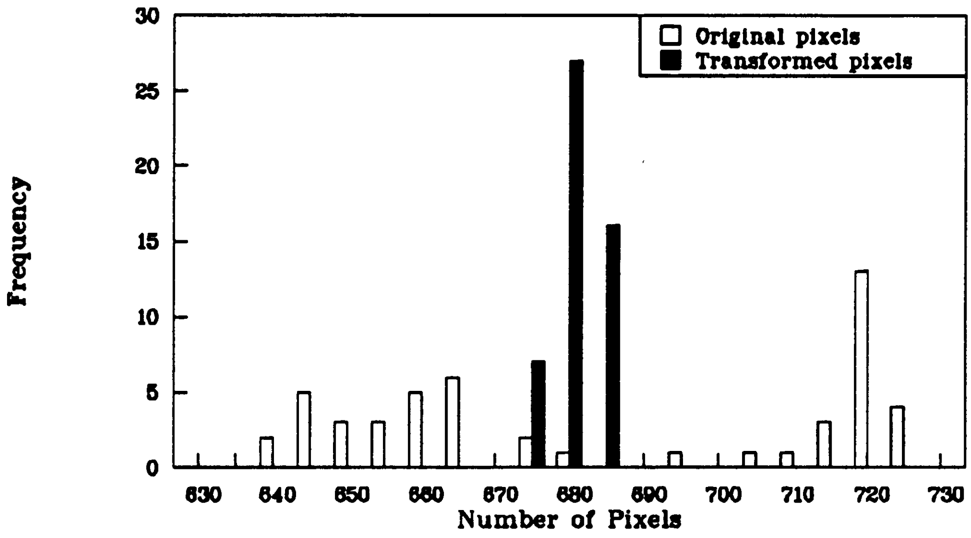


Figure 17 Distribution of Pixel Readings for 50 Images of one Object

It seemed logical to use the recorded number of pixels of the reference target as a means to "correct" the pixel count for measured objects. Higher or lower than average number of pixels for the reference target should result in a percentage addition or subtraction from the originally calculated pixel count for the measured objects. The transformation equation for the left and the right targets used are:

$$CPixL = \frac{PDevL * (MeanRef - Ref) * PixL}{PDevRef * MeanRef} + PixL \qquad [5]$$

$$CPixR = \frac{PDevR * (MeanRef - Ref) * PixR}{PDevRef * MeanRef} + PixR \qquad [6]$$

where:

CPixL = Corrected pixel count for left side target

CPixR = Corrected pixel count for right side target

PDevL = Percentage standard deviation for left side target, constant: 2.3

PDevR = Percentage standard deviation for right side target, constant: 2.8

PDevRef = Percentage standard deviation for reference target, constant: 3.5

MeanRef = Mean of reference target pixel count, constant: 776

PixL = Pixel count for left side target

PixR = Pixel count for right side target

Ref = Pixel count for reference target

The above equations transformed the bimodal distributions into normal distributions and reduced the percentage standard deviation from approximately 2.5 % to below 0.5 % as shown for the example distribution in Figure 17 and in Appendix D (*Uncorrected vs. Corrected Pixel Data*). All other tests with metal targets were performed with this correction.

5.2 Object Orientation and Location in Field of View

The orientation of the gray targets with its sharply defined edges could have an influence on pixels counted. A test to determine if there is any influence of target orientation was performed. Two medium sized targets of approximately 17 cm^2 were placed one on each side of the conveyor belt (18 cm apart). Four different orientations were chosen as shown in Figure 18. The targets in those pictures are identical. Their appearance is distorted because the rectangular field of view is printed square. Ten images were taken of each target orientation (Appendix D. *Pixel Measurements for Four Different Orientation of Metal Targets*). A one-way analysis of variance (ANOVA) was employed to test if the means of each orientation were significantly different (Ott, 1988). The test statistic used was an F-test which expresses the ratio between the variance due to different locations and the variance due to experimental variation or error. The results are shown in Table 4. Based on the F-test it was concluded that the differences between means of those four orientations were not significant at a 5 % error level.

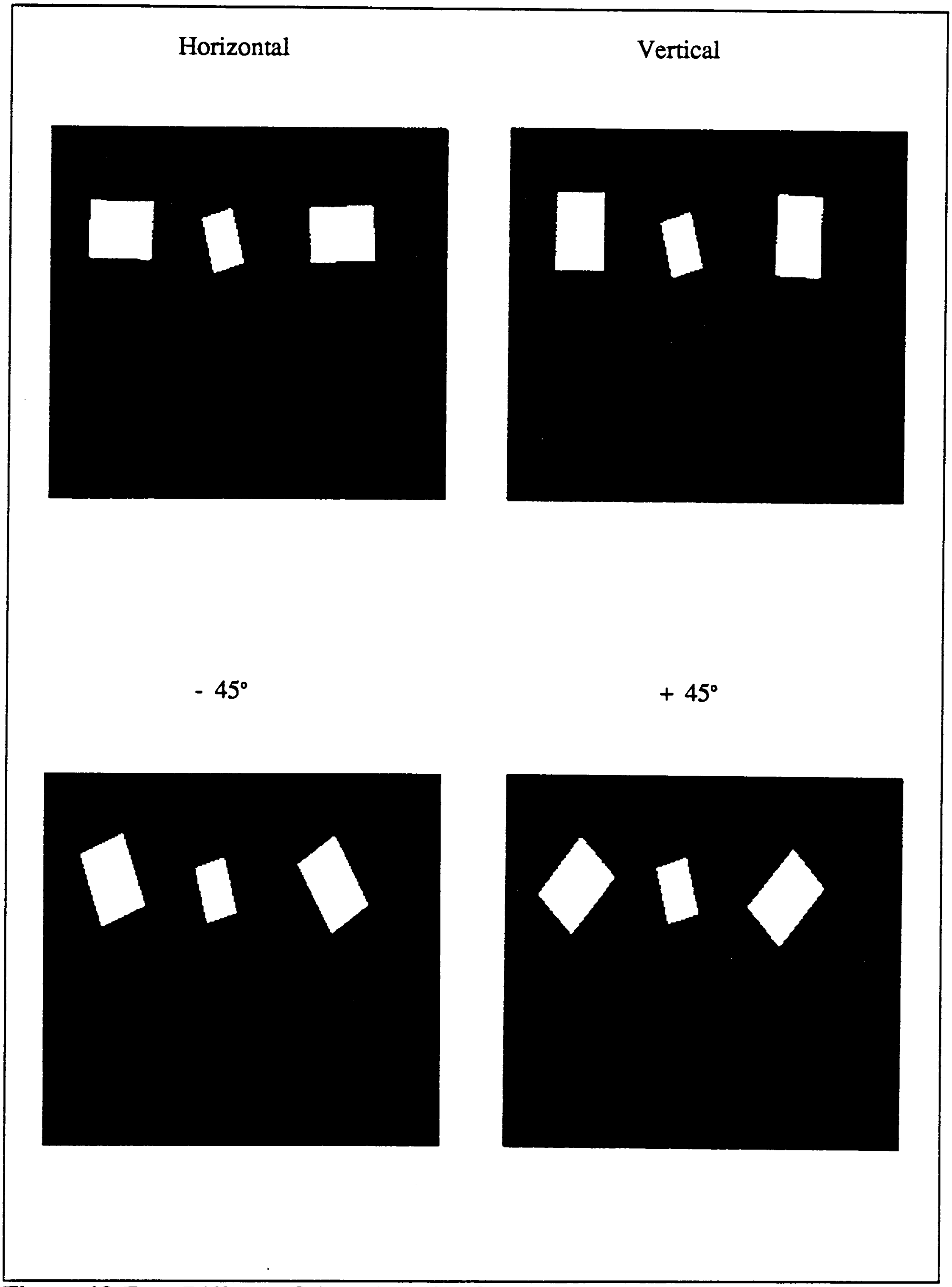


Figure 18 Four Different Orientations for Gray Rectangular Targets

Table 4 Comparison of Number of Pixels for Four Target Orientations

Target orientation	horizontal	vertical	+ 45°	- 45°
Target area : 1717mm²	(Left side of belt)			
Observations	10	10	10	10
Mean [pixel]	1560.6	1558.9	1560.3	1554.9
Standard deviation [pixel]	7.10	6.57	6.92	8.73
Target area : 1703 mm²	(Right side of the belt)			
Observations	10	10	10	10
Mean [pixel]	1551.5	1547.4	1549.1	1550.8
Standard deviation [pixel]	7.82	5.79	7.95	8.22

F-statistic :

Null Hypothesis, H_0 : $\mu_1 = \mu_2 = \mu_3 = \mu_4$
Alternative Hypothesis, H_1 : At least one pair of means is unequal

1717 mm² target area

Test Statistic, TS : F(obs) = 1.26
Rejection Region, RR : F(obs) > F(critical) = F(3,39) = 2.84

Conclusion : Since F(obs) < F(critical) at $\alpha = 5$ % ==> Accept H_0

1703 mm² target area

Test Statistic, TS : F(obs) = 0.60
Rejection Region, RR : F(obs) > F(critical) = F(3,39) = 2.84

Conclusion : Since F(obs) < F(critical) at $\alpha = 5$ % ==> Accept H_0

The pixel counted for the same targets could be different depending on which side of the picture they appear. Six different targets covering the available size range from 4.7 cm^2 to 28 cm^2 were placed on one side of the conveyor belt and 10 pictures were taken of each. Then they were placed on the other side of the belt and another 10 pictures were taken (Appendix D. *Data for Comparison of Measurements on the Left and Right Side of the Belt*). Each of the pixel means measured on the left side were compared to the corresponding mean measured on the right side through a t-test. Comparing the means with a t-test is only valid if the variances can be assumed equal. An F-test performed to compare variances showed that the variances were equal except for the 17.17 cm^2 target. In this case an approximate t-test or Welch test is appropriate. Test results for the comparison of means are shown in Table 5. There is a significant difference between the means measured on each side except for the smallest target. The variations between measurements on one side are larger than the variation caused by different location. This can be explained through the larger ratio between circumference and area for smaller targets. Variations in pixel count occur predominantly close to the circumference of objects which explains the higher variation in pixel count for smaller objects.

The discovery of differences in measurements between the two locations raised the question if there were any differences between locations on one side of the belt. A 4 to 5 cm variation in position on each side of the belt estimated because a singulation device can not place the oyster meats in exact the same position. The same test as above was conducted (Appendix D. *Data for Comparison of Measurements on one Side*

Table 5 Comparison of Pixel Readings for the Left and Right Side

t-statistic :

Null Hypothesis, H_0	:	$\mu_1 = \mu_2$
Alternative Hypothesis, H_1	:	$\mu_1 <> \mu_2$
Test Statistic, TS	:	t(obs)
Rejection Region, RR	:	abs[t(obs)] > t(critical) = $t(18,\alpha=5\%)$

Target area [mm²]	Mean [pixel]	Std. dev. [pixel]	t(obs)	t(critical) $t(18,\alpha=5\%)$	Conclusion
LEFT SIDE OF BELT					
3123	2883.7	8.19	-	-	-
2806	2581.4	11.88	-	-	-
1717	1567.1	4.86	-	-	-
1703	1565.5	5.61	-	-	-
1564	1431.1	7.41	-	-	-
471	428.5	9.19	-	-	-
RIGHT SIDE OF BELT					
3123	2807.7	8.19	9.26	1.734	reject H_0
2806	2563.6	9.87	3.64	1.734	reject H_0
1717	1534.6	16.99	5.82*	1.734	reject H_0
1703	1535.7	8.91	8.95	1.734	reject H_0
1564	1421.6	6.46	3.05	1.734	reject H_0
471	422.4	11.56	1.30	1.734	accept H_0

* t(obs) under the assumption of unequal variances

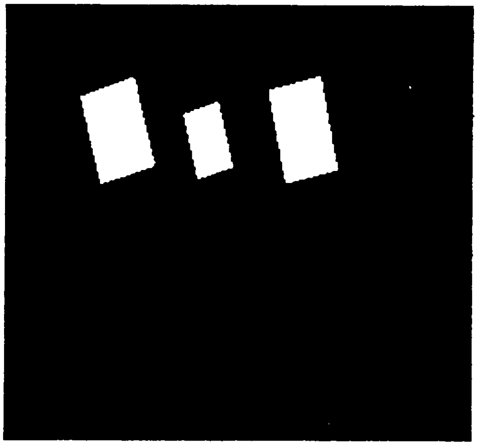
f the Belt). But this time the targets were placed in two extreme locations on one side of the belt, close to the middle axis and far from it as shown in Figure 19. The results of the comparison of the means are shown in Table 6. Under the assumption of equal variances there is no significant influence of varied target position on one side of the belt.

The detected measurement differences between the two sides of the belt are no problem if the calibration is done separately for each side of the belt which is described in Section 5.4 and 5.5.

5.3 Object Velocity

It is well known that flash exposure of moving objects eliminates blurred images through the very short exposure times. But to make sure that this is really happening a series of measurements using three different speeds were performed. Ten pictures of two objects were taken under stationary condition and approximate working belt speed of 0.3 and 0.4 m/s (Appendix D. *Pixel Measurements in Three Different Speeds*). The results shown in Table 7 indicate no statistical difference in pixel area measurement at the 5% confidence level. The area measurements are speed independent.

7 cm from the center axis of the belt



11 cm from the center axis of the belt

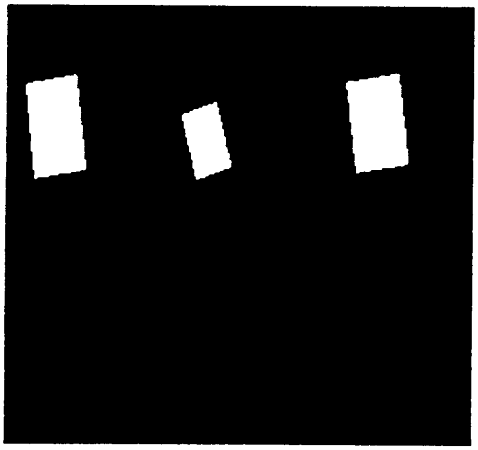


Figure 19 Test Locations for Measurements on one Side of the Belt

Table 6 Comparison of Pixel Readings on One Side of the Belt

Position 1 : 11 cm from the center axis of the belt
Position 2 : 7 cm from the center axis of the belt

Target position	1	2
Target area : 1717mm² (Left side of belt)		
Observations	10	10
Mean [pixel]	1570.2	1568.7
Standard deviation [pixel]	8.08	7.38
Target area : 1703 mm² (Right side of the belt)		
Observations	10	10
Mean [pixel]	1552.8	1554.9
Standard deviation [pixel]	8.18	7.44

t-statistic :

Null Hypothesis, H_0 : $\mu_1 = \mu_2$
Alternative Hypothesis, H_1 : $\mu_1 <> \mu_2$

1717 mm² target area

Test Statistic, TS : t(obs) = 0.4247
Rejection Region, RR : abs[t(obs)] > t(critical) = t(18) = 1.734

Conclusion : Since t(obs) < t(critical) at α = 5 % ==> Accept H_0

1703 mm² target area

Test Statistic, TS : t(obs) = - 0.5689
Rejection Region, RR : abs[t(obs)] > F(critical) = F(18) = 1.734

Conclusion : Since abs[t(obs)] < t(critical) at α = 5 % ==> Accept H_0

Table 7 Area Measurements at Three Object Velocities

	Stationary	Speed 0.3 m/s	0.4 m/s
Target area : 1027mm²	(Left side of belt)		
Observations	15	15	15
Mean [pixel]	905.2	904.4	905.6
Standard deviation [pixel]	4.36	4.79	4.77
Target area : 2207 mm²	(Right side of the belt)		
Observations	15	15	15
Mean [pixel]	1985.0	1985.7	1987.4
Standard deviation [pixel]	6.79	7.62	7.66

F-statistic

Null Hypothesis, H_0 : $\mu_1 = \mu_2 = \mu_3$
Alternative Hypothesis, H_1 : At least one pair of means is unequal

1027 mm² target area

Test Statistic, TS : F(obs) = 0.28
Rejection Region, RR : F(obs) > F(critical) = F(2,44) = 3.23

Conclusion : Since F(obs) < F(critical) at $\alpha = 5$ % ==> Accept H_0

2207 mm² target area

Test Statistic, TS : F(obs) = 0.41
Rejection Region, RR : F(obs) > F(critical) = F(2,44) = 3.23

Conclusion : Since F(obs) < F(critical) at $\alpha = 5$ % ==> Accept H_0

5.4 Calibration for Pixel vs. Area

The vision system was calibrated with 26 gray-colored rectangular metal targets of known area to measure projected object area. Target sizes covered the range of oyster sizes recorded by Awa (1988), which were between 5 and almost 30 cm^2. A linear regression with and without intercept was performed. It was logical to assume that an object of zero area would have no pixel counted. It was observed, however, that very small objects would not always appear in the digitized picture, resulting in a negative intercept for a linear regression with intercept. Table 8 shows the results of the statistical analysis. Appendix D (*Area Calibration Data for 26 Metal Targets*) contains the measured areas and pixel counts. The negative intercepts were -13.3 and -23.5 mm^2 for the left and right side, respectively. A t-test to determine whether the intercepts were statistically significant or should be dropped was performed. The test led to the conclusion that the null hypothesis (intercept = 0) should be accepted at the 5% significance level. The final calibration equation for area measurement were:

$$A_r = 1.09864 * P_r \quad [7]$$

$$A_l = 1.09053 * P_l \quad [8]$$

where:

A = area in mm^2,

P = number of pixels, and subscripts r = right side, l = left side

Table 8 Results of Linear Regression for Pixel vs. Area of Metal Targets

	LEFT Side		RIGHT Side	
Intercept	No	Yes	No	Yes
Sum of Squares Error (SSE)	18989.96	18317.84	36023.61	34085.84
Mean Square Error (MSE)	759.598	763.24	1440.94	1420.24
Degrees of Freedom (DF)	26	25	26	25
Root Mean Square Error	27.561	27.626	37.96	37.68
Intercept	-	- 13.3057	-	- 23.5083
Standard Error	-	14.179	-	20.1257
Slope	1.09053	1.09774	1.09864	1.11194
Standard Error	0.00317	0.00831	0.00453	0.01224
R^2	0.99978*	0.99863	0.99957*	0.99710
Sum of Residuals	- 50.51	0	- 82.43	0

* R^2 is not the same measure of explained variation for a regression without intercept than for one with intercept

t-statistic :

Null Hypothesis, H_0 : Intercept = 0
Alternative Hypothesis, H_1 : Intercept <> 0

Test Statistic, TS : t(obs) = Intercept / Std. Error Intercept

Rejection Region, RR : t(obs) > t(critical)
t(critical) = $t(24,\alpha=5\%)$ = 1.711

Model for LEFT side of the belt

Test Statistic, TS : t(obs) = 0.938
Conclusion : Since t(obs) < t(critical) at $\alpha = 5\ \%$ ==> Accept H_0

Model for RIGHT side of the belt

Test Statistic, TS : t(obs) = 1.168
Conclusion : Since t(obs) < t(critical) at $\alpha = 5\ \%$ ==> Accept H_0

with an estimated error of $\pm$ 0.76 cm^2 for the right side equation, and a $\pm$ 0.55 cm^2 for the left side equation. The measurement error would normally be reported at the 95% confidence interval for predictions from the measured number of pixels. This is not applicable for a regression equation with forced zero intercept (Ott, 1988). Another measure of variability of the model would be the mean square error. The root mean square error would be the sample standard deviation due to experimental error. Two standard deviations are roughly equal to a 95% confidence level. Therefore, the root mean square error was doubled to give the error estimate mentioned above. Figure 20 and Figure 21 show the linear area/pixel relationship for the left and right side of the vision field. Both figures graphically demonstrate the low measurement error. But the following test pictures showed that oyster meats could not be measured with the determined equations, because varying oyster meat color caused some image deterioration. The result was missorting, requiring vision system recalibration as described in the next section.

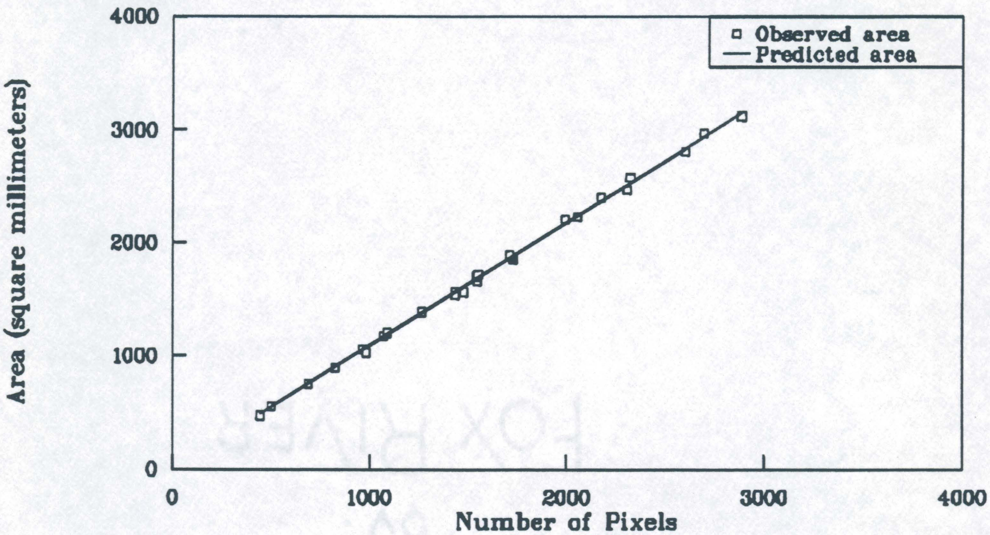


Figure 20 Linear Regression for Pixel vs. Area on Left Side of the Belt

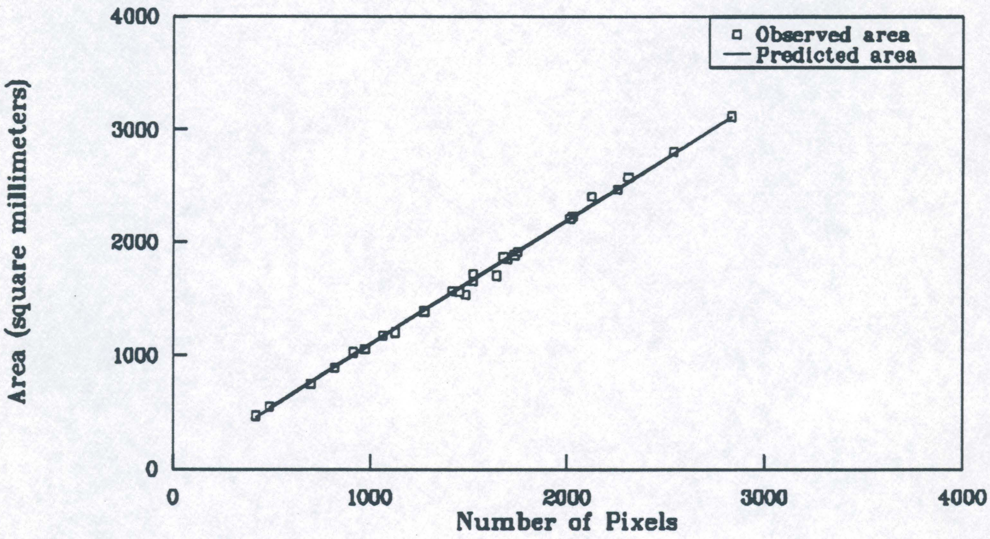


Figure 21 Linear Regression for Pixel vs. Area on Right Side of the Belt

5.5 Calibration for Pixel vs. Weight

System calibration for oyster meats required some readjustment of the vision system. Oyster meats are not uniform enough in color to allow the high threshold of a 0.95 V video signal as a threshold for white (gray level 64). Readjusting the black and white levels through a voltage threshold was not possible without major deterioration of the left oyster meat image (Figure 22). When the voltage threshold was lowered to obtain a complete image of the left meat, the right meat image was completely overexposed resulting in loss of contours and "bloomed" images. It was observed that the right side of the picture always exhibited higher gray level pixels, with bright pixels continually appearing along the right edge of each picture (Figure 22). Moving the light source to different angles of incidence, or to the opposite side of the objects did not alter this phenomenon. The camera was rotated 180° about the lens axis to determine if uneven coating on the camera sensor might be the cause of the higher gray level pixels. The bright pixels appeared again on the right side of the picture, even though the imaged object had switched position in the picture. This indicated that either the sensor or the digitizer may cause the unevenly exposed pictures. The bright spot is of different size for each picture containing gray levels as high as oyster meat gray levels. In approximately 10% of all pictures the white spot touches the oyster image and is inseparable from the meat through thresholding. The oyster meat will inevitably be recorded larger than its actual size.

As the left side of the picture appeared more evenly exposed, the camera was rotated

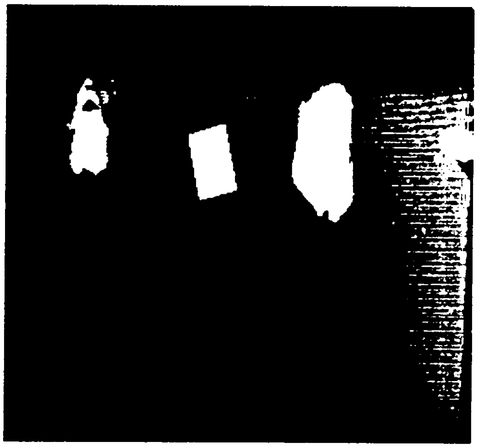


Figure 22 Deteriorated Oyster Meat Images

90° about the lens axis and moved laterally to place the oyster meat images at the left of the picture (Figure 23). The lens object distance was increased to 61 cm to the same field of view width as before. Black and white level voltages were readjusted to lower levels (black: 0.77 V; white: 0.9 V). The left half of the picture was divided into three horizontal sections (line 0-95, section 1; line 96-150, section 2; line 151-243, section 3) and histograms of each section were studied. The images are still unevenly exposed, however, not as bad as on the other side of the picture. The upper section is not as bright as the lower section. The exposure of the lower section could be controlled through an additional threshold of gray level 27 to separate the oyster meats. No threshold was needed for the upper part of the picture. The area around the reference target was thresholded with 20. Those three threshold level were determined experimentally, testing different levels with 20 different oyster pictures. Figure 23 shows the result of the different thresholds for a typical picture.

Optimal light and exposure conditions were achieved with a flash intensity of 130 lux at the object and a f/5.6 aperture. The reference target area in number of pixel averaged over 30 readings was 655 $\pm$10 pixels with this setting. Correlation coefficient analyses as described in Section 5.1 showed a low correlation coefficient between the reference target and the oyster meat area for both sides. Appendix D (*Uncorrected Pixel Data for Oyster Meats*) shows one typical data set of 30 pixel readings. The correlation coefficient between pixel readings for reference target and right side oyster meat (threshold 27) was only $r = 0.32$, between reference target and left side meat r was equals 0.70 (threshold 0). The correction equations [5] and [6] are not valid under

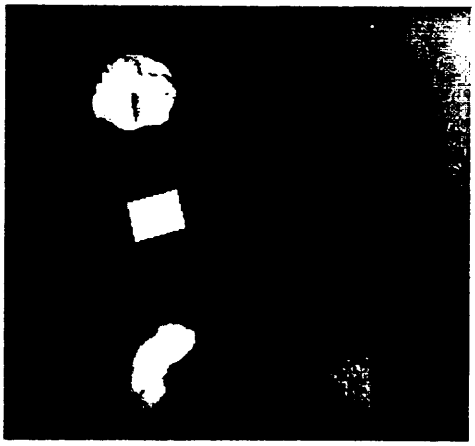


Figure 23 Oyster Meat Images

these conditions. They were not used for the subsequent calibration. The calibration was performed with the actually calculated pixel data.

A quart of *Standards* and a quart of *Selects* were used to calibrate the vision system. The meats were mixed, with individuals chosen arbitrarily. Each side of the belt was calibrated separately with 54 oyster meats (27 on each side). The oyster meats were first weighed, then dropped onto the conveyor from a glass plate 10 cm above the belt simulating the placement of meats from a singulator. The meats were conveyed through the vision field where their images were taken. The number of pixels and the weight of each oyster were recorded on diskette. The calibration data are listed in Appendix D (*Weight/Area Calibration Data for 54 Oyster Meats*).

A linear regression with and without intercept were performed using weight as the independent variable and number of pixels as the dependent variable. Table 6 shows the linear regression results. The R^2 value of approximately 0.9 indicates the higher variability of the biological object. A t-test on the significance of the intercept showed that the intercept differs from zero at a 95% confidence level. The final calibration equation for weight measurement were:

$$W_r = -2.5153 + 0.01140 * P_r \quad [9]$$

$$W_l = -1.5275 + 0.009582 * P_l \quad [10]$$

Table 9 Results for Linear Regression Oyster Weight vs. Image Pixel

	LEFT Side		RIGHT Side	
Intercept	No	Yes	No	Yes
Sum of Squares Error (SSE)	3235.174	244.4078	4006.948	322.2981
Mean Square Error (MSE)	1.1607	1.059	1.8757	1.2569
Degrees of Freedom (DF)	27	26	27	26
Root Mean Square Error	1.07734	1.0293	1.3695	1.2527
Intercept	-	- 1.52754	-	- 2.5153
Standard Error	-	0.8182	-	1.0206
Slope	0.00839	0.009582	0.009498	0.01140
Standard Error	0.000159	0.000631	0.000205	0.00079
R^2	0.99075[x]	0.90223	0.98797[x]	0.89147
Sum of Residuals	- 2.41	0	- 3.78	0

[x] R^2 is not the same measure of explained variation for a regression without intercept than for one with intercept

t-statistic :

Null Hypothesis, H_0 : Intercept = 0
Alternative Hypothesis, H_1 : Intercept <> 0

Test Statistic, TS : t(obs) = Intercept / Std. Error Intercept

Rejection Region, RR : abs[t(obs)] > t(critical)
t(critical) = $t(25,\alpha=5\%)$ = 1.708

Model for LEFT side of the belt

Test Statistic, TS : t(obs) = - 1.87
Conclusion : Since abs[t(obs)] > t(critical) at α = 5 % ==> Reject H_0

Model for RIGHT side of the belt

Test Statistic, TS : t(obs) = - 2.46
Conclusion : Since abs[t(obs)] > t(critical) at α = 5 % ==> Reject H_0

where:

W = weight in g,

P = number of pixels, and subscripts r = right side, l = left side

with an estimated prediction error of $\pm$ 2.8 g for the right side equation, and a $\pm$ 2.41 g for the left side equation. The estimated errors are the prediction interval limits in the prediction range between 5 and 17 g for oyster meat weight. The data and the model used are illustrated in Figure 24 for the left side and Figure 25 for the right side of the belt.

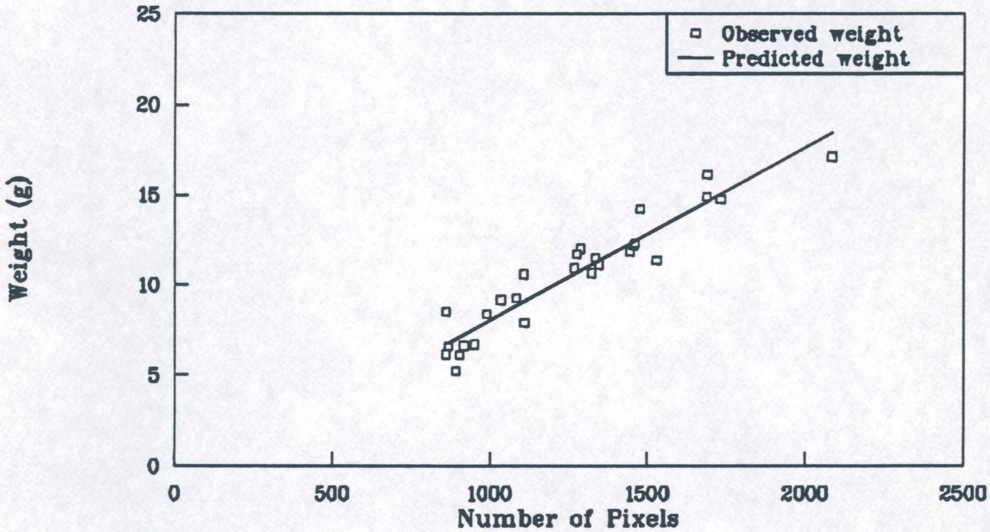


Figure 24 Linear Regression for Oyster Meat Area vs. Weight on Left Side of the Belt

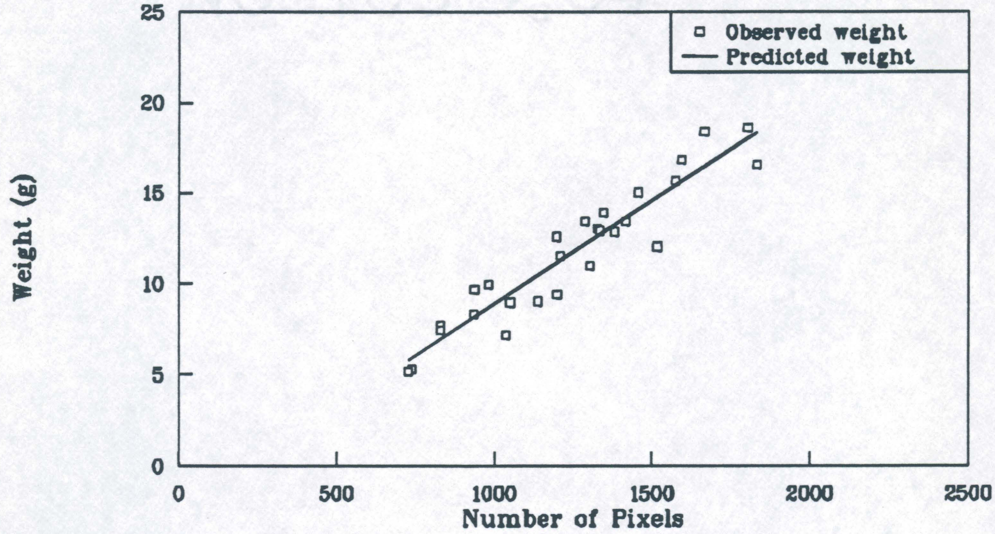


Figure 25 Linear Regression for Oyster Meat Area vs. Weight on Right Side of the Belt

6. Sorting Performance

After calibration of the vision system a sample of oyster meats were sorted. The values of prediction equations [9] and [10] were incorporated into the sorting program. The belt speed was set at 0.15 m/s to permit the vision system to grade the oyster meats before they reached the first jet nozzle. High data transmission time did not allow a higher belt speed. To permit consecutive sorting the distances between each oyster on the belt could not be less than 60 cm. Each meat was weighed before machine sorting. Knowing the precise weight of each oyster meat allowed evaluation of the machine's sorting quality. A quart of *Standards* oyster meats mixed with a quart of *Selects*, from which 60 oyster meats were arbitrarily chosen and sorted. Half were sorted on the left side and the other half were sorted on the right side of the belt. The sorting results for each side are listed in Table 10 and Table 11. The sorting results for all 60 oyster meats combined are shown in Figure 26. The vertical lines indicate limits between classes according to their actually observed weight. The left region contains *Counts*, the right region belongs to *Selects* leaving the middle for *Standards*. The horizontal lines do the same for the predicted weights with the *Counts* at the bottom of the chart and the *Select* oyster meats at the top.

The vision system graded four meats on the left side and three meats on the right side into improper categories -- totaling 12.5% missorted meats. Correct classified oyster meats are located in the boxes on the left to right ascending diagonal in Figure 26. Observations shown in other boxes are misclassified. Figure 26 shows that errors occur only near the cut-off between two sizes and were half too small and half too large. A certain percentage of misclassification will always occur due to the wide variety of the biological material as heavy oysters can have medium projected area, because they are thicker than the average. The natural variation in optical properties (color shades) are another source of error. One error was introduced through the vision system's shifting exposure of pictures (subsequent images of the same unmoved object have different pixel counts), and the statistical error in the grading equations adds a fourth source of error. The vision system related error can be reduced through replacement of the current system with a more sophisticated system.

Table 10 Sorting Performance for 30 Oyster Meats on the Left Side

Meas. No.	Observed Weight [g]	Predicted Weight [g]	GRADED Observed	GRADED Predicted
1	10.853	11.648	Standard	Standard
2	17.466	15.308	Select	Select
3	7.708	9.894	Standard	Standard
4	9.651	9.453	Standard	Standard
5	18.879	8.461	Select	Select
6	12.564	9.338	Standard	Standard
7	10.224	10.613	Standard	Standard
8	5.635	5.860	Count	Count
9	10.347	12.845	Standard	Standard
10	16.121	18.317	Select	Select
11	9.092	12.462	Standard	Standard
12	6.384	5.764	Count	Count
13	4.482	6.061	Count	Count
14	14.523	13.823	Select	Select
15	12.602	13.976	**Standard**	**Select**
16	7.701	8.658	Standard	Standard
17	11.085	10.718	Standard	Standard
18	4.796	5.007	Count	Count
19	9.602	7.192	**Standard**	**Count**
20	12.619	11.600	Standard	Standard
21	19.099	16.717	Select	Select
22	12.703	12.759	Standard	Standard
23	9.837	9.722	Standard	Standard
24	12.583	13.670	**Standard**	**Select**
25	7.810	10.632	Standard	Standard
26	11.894	9.904	Standard	Standard
27	12.564	11.149	Standard	Standard
28	7.583	6.330	Count	Count
29	12.671	12.960	Standard	Standard
30	13.611	12.635	**Select**	**Standard**

Table 11 Sorting Performance for 30 Oyster Meats on the Right Side

Meas. No.	Observed Weight [g]	Predicted Weight [g]	GRADED Observed	GRADED Predicted
1	10.864	10.040	Standard	Standard
2	11.967	10.405	Standard	Standard
3	14.592	16.267	Select	Select
4	18.230	15.571	Select	Select
5	11.182	13.325	**Standard**	**Select**
6	8.671	7.725	Standard	Standard
7	12.637	12.333	Standard	Standard
8	14.863	12.618	Standard	Standard
9	3.824	5.433	Count	Count
10	10.781	11.890	Standard	Standard
11	6.347	6.996	Count	Count
12	7.920	7.976	Standard	Standard
13	4.734	5.490	Count	Count
14	11.182	13.325	**Standard**	**Select**
15	15.891	15.742	Select	Select
16	15.649	14.625	Select	Select
17	12.011	11.090	Standard	Standard
18	19.587	17.544	Select	Select
19	13.845	13.656	Select	Select
20	7.672	9.801	Standard	Standard
21	6.254	6.744	Count	Count
22	12.369	9.584	Standard	Standard
23	15.703	15.298	Select	Select
24	17.002	17.442	Select	Select
25	10.153	10.873	Standard	Standard
26	7.393	10.303	**Count**	**Standard**
27	9.684	9.653	Standard	Standard
28	13.781	14.556	Select	Select
29	17.184	19.722	Select	Select
30	22.239	20.680	Select	Select

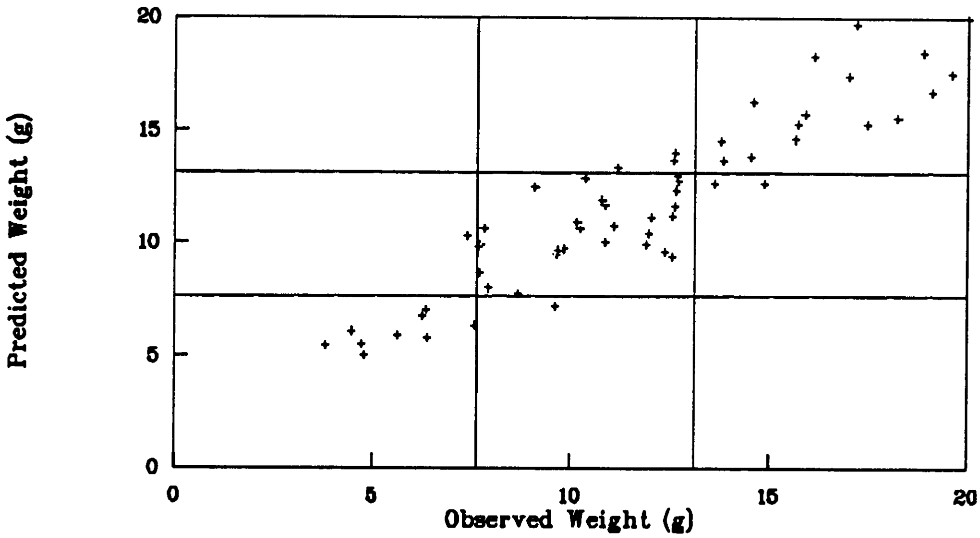


Figure 26 Sorting Results for a Total of 60 Oyster Meats

7. Summary and Conclusions

The objective of this study was to design and test a machine that grades and sorts oyster meats at a rate of 5 oyster meats/second according to their size.

The sorting machine consists of a dark conveyor belt transporting 2 lines of singulated oyster meats through a machine vision grading station and then along a row of fast acting water jet valves which separates the stream of oysters into 3 classes. The vision system consists of a monochrome television camera, strobe light illumination to "freeze" the images, a digitizer/transmitter and a personal computer as image processing unit. Software synchronizes the flash light and digitization of images and calculates projected area of each meat using the planimeter method. Then it grades the meats into three categories, *Selects* (large), *Standards* (medium), and *Counts* (small). The conveyor moves the oysters past a row of high-impact water jets that spray the oyster meats off the belt into different containers beneath the belt. The grading results are sent to a valve control board before each meat reaches the first spray valve. The control board monitors the position of all meats past the vision station and actuates the spray valves as they reach the appropriate valves.

Several tests were conducted to determine which factors influence the projected area measurement by the vision system. Statistical analysis showed that area measurements in number of pixels is independent from the object orientation, but is dependent upon the object position in the picture. Objects on the left side of the belt appear larger than on the right side. Therefore, system calibration was conducted separately for each side. The measurements were independent from the speed they moved by the vision system.

The vision system was calibrated for actual area measurement with gray-colored metal targets of known area. The models for both sides were linear with intercept zero. The estimated errors were $\pm$ 0.76 cm^2 for the right side equation, and a $\pm$ 0.55 cm^2 for the left side equation for an estimation range between 5 and 30 cm^2.

The vision system required readjustment because oyster meats are not uniform in color and appear various shades of gray and beige. The calibration after adjustment was done using the relationship between projected area and oyster meat weight to avoid the elaborate procedure of measuring the actual area of each oyster meat. Weight can be used as a grading criterion because of its very high correlation to volume. Each side of the belt was calibrated separately with 54 oyster meats (27 on each side). The models used were two linear equations (one for each side) with non-zero intercepts. Estimated prediction errors were $\pm$ 2.8 g for the right side equation, and a $\pm$ 2.41 g for the left side equation for an estimation range between 5 and 17 g.

Sixty oyster meats were sorted in a final sorting test. They were chosen arbitrarily from commercially graded *Standards* and *Selects* covering the whole range of sizes available for each side of the belt. Each oyster weight was recorded before it was graded and sorted by the machine. 87.5% of all oyster meats were sorted correctly. Errors in sorting were only made near the cut-off limits between classes, which will inevitably occur because of the high variability the biological product. It can be concluded that the sorting machine improves the sorting quality substantially compared to human grading where 50 to 80% of oyster meats were found to be out of grade in samples from three commercial processors.

One grading cycle time period requires approximately 3.2 s, resulting in a grading rate of 37 oyster meats/min, which is far below the desired rate of 300 meats/min. This was due to the speed of the image processing components. Picture transmission accounts for more than 95% of the cycle time. The software and the spray sorting system, however, perform at the desired rate. Incorporation of parallel data transfer digitizer/transmitter board with a high speed transmission rate should allow the desired grading rate.

8. Recommendations

The work presented by this thesis shows that it is possible to sort oyster meats by machine vision with significantly higher accuracy than human sorting. Following are recommendations for future development of the oyster meat sorting machine.

Grading rate can be improved with a high-speed image digitizer/transmitter board. The electronic flash light source should be replaced with a higher speed strobe light designed for applications with machine vision systems. Such strobe lights are specially designed for even light distribution and low variation in discharged energy between flashes, which should upgrade picture, and therefore sorting, quality. The camera should be replaced with a unit possessing a higher quality, light-sensitive sensor to decrease variation between pictures caused by either shifting sensor response or uneven coating. A solid-state camera would be a good choice. Once a higher quality vision system is incorporated, a smaller illumination box could be designed to accommodate both camera and lighting equipment. A different vision system will call some software adaptation. The picture receiver subroutine and the picture format (pixel array) has to be changed, but the software core can remain the same. The author believes that after the suggested changes have been made to the machine, it will sort at the desired rate of 5 oyster meats/s.

This portion of the desired mechanized oyster meat sorting system is still in the experimental stage, but will ultimately be used in a production environment. At that point all electronic components, camera, and lighting system will have to be protected against intrusions of salt, humidity, and possible water splash.

Finally, to complete the system, a singulation device will have to be designed, built, and tested. The singulator will have to rapidly set oyster meats side-by-side on the belt of the machine sorting device described in this thesis. Depending on singulator design, the machine vision system may have to be modified.

References

Awa, T. W., 1988. Development of an Inexpensive Computer Vision System for Grading Oyster Meats. Unpublished Masters Thesis, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Awa, T. W., R. K. Byler, and K. C. Diehl, 1988. Development of an Inexpensive Oyster Meats Grader. ASAE Paper No. 88-3539, St. Joseph, Michigan.

Batchelor, B. G., D. A. Hill, and D. C. Hodgson, 1985. Automated Visual Inspection. IFS Publications Ltd., Bedford, UK.

Berlage, A. G., T. M. Cooper, and J. F. Aristazabal, 1988. Machine Vision Identification of Diploid and Tetraploid Ryegrass Seed. TRANSACTIONS of the ASAE, 31(1):24-27.

Chen, S. S., and F. W. Wheaton, 1987. Oyster Hinge Line Detection Using Image Processing. ASAE Paper No. 87-3047, St. Joseph, Michigan.

Ciarcia, S. A., 1987a. Build a Grayscale Video Digitizer, Part 1: Display/Receiver. Byte, 12(5):95-138.

Ciarcia, S. A., 1987b. Build a Grayscale Video Digitizer, Part 2: Digitizer/Transmitter. Byte, 12(6):129-138.

Ciarcia, S. A., 1987c. Using the IMAGEWISE Video Digitizer, Part 1:Image Processing Byte, 12(7):113-119.

Cielo, P., 1988. Optical Techniques for Industrial Inspection. Academic Press, Inc., 1250 Sixth Avenue, Dan Diego, CA 92101.

Delwiche, M. J., S. Tang, and J. F. Thomson, 1988. Prune Defect Detection by Line-Scan Imaging, ASAE Paper No. 88-3024, St. Joseph, Michigan.

Diehl, K. C., T. W. Awa, R. K. Byler, M. van Gelder, and M. B. Koslav, 1988. Geometric Properties of Oyster Meat as Related to Sorting. ASAE Paper No. 88-6526, St. Joseph, Michigan.

Faugeras, O. D., 1983. Fundamentals in Computer Vision. Cambridge University Press, Cambridge, UK.

Food and Drug Administration, 1988. Requirements for Specific Standardized Fish and Shellfish. Code of Federal Regulations, Title 21, Part 161, Sections 161.130-161.140.

Gavenda, B., 1989. Principles of Strobe Lighting. Lighting and Optic Technology Clinic, June 13, 1989, Nashville, Tennessee.

Gonzalez, R. C., R. E. Woods, and W. T. Swain, 1986. Digital Image Processing: An Introduction. Digital Design, xx(xx) march 25:15-20.

Goshorn, L. A., 1983. Vision Systems Eye Real-Time Speeds through Multiprocessor Architecture. Electronics, 56(25):137-143.

Gunasekaran, S., T. M. Cooper, A. G. Berlage, and P. Krishnan, 1988. Image Processing for Stress Cracks in Corn Kernels, TRANSACTIONS of the ASAE, 30(1):266-271.

Hayes, T., 1988. Personal Communication. Coast Oyster Co., POB 166, South Bend, Washington.

Jameco Electronics, 1988. User Manual for jameco Model JE1017 Microcomputer. 1355 Shoreway Road, Belmont, CA 94002.

Kranzler, G. A., 1984. Digital Image Processing for Agriculture. ASAE Paper No. 84-5513, St. Joseph, Michigan.

Lapidus, S. N., and A. C. Englander, 1985. Understanding How Images are Digitized. Vision '85, Conference Proceedings, March 25-28, 1985, Detroit, MI, pp. 4-22 - 4-33.

Marchant, J. A., C. M. Onyango, and M. J. Street, 1988. High Speed Sorting of Potatoes Using Computer Vision, ASAE Paper No. 88-3540, St. Joseph, Michigan.

Matek, W., D. Muhs and H. Wittel, 1983. Roloff/Matek Maschinenelemente. Friedr. Vieweg & Sohn, Braunschweig/Wiesbaden, FRG.

Mersch, S. H., 1984. Polarized Lighting for Machine Vision Applications. Third Annual Applied Machine Vision Conference Proceedings, Feb. 27 - March 1, 1984, Schaumburg, IL, pp. 4-40 - 4-55.

Micromint, Inc., 1987. *ImageWise* Digitizer/Transmitter Users Manual. 4 Park Street, Vernon, CT 06066.

Nakahara, S., A. Maeda, and Y. Nomura, 1979. Automated Cucumber Sorting Using Pattern Recognition Technique. IEEE, Proc. Denshi Tokyo, No. 18:46-48.

Ott, L., 1988. An Introduction to Statistical Methods and Data Analysis. PWS Publishing Company, Boston, 3rd Edition.

Site, P. W., and M. J. Delwiche, 1988. Computer Vision to Locate Fruit on a Tree. TRANSACTIONS of the ASAE, 31(1):257-263.

Smith, J. D., 1971. Development of Improved Techniques for Shucking Pacific Oysters. Publication No. WSG 71-1, Division of Marine Resources, University of Washington, Seattle, Washington, pp. 33.

Smith, D. K., 1972. Package Conveyors: Design and Estimation. Charles Griffin & Company Ltd, London, 1. Ed., p. 19 ff.

Somers, J. D., and E. Frank, 1987. Optical Sorting, Food Proccessing, 48(1)78-79.

Spady, O. A., 1988. Personal Communication. Battery Park Oyster, Battery Park, VA

Tojeiro, P., and F. W. Wheaton, 1987. Computer Vision Applied to Oyster Orientation. ASAE Paper No. 87-3046, St. Joseph, Michigan.

Wheaton, F. W., 1974. Oyster Shucking with Infrared Radiation. TRANSACTIONS of the ASAE, 17(2):342-345.

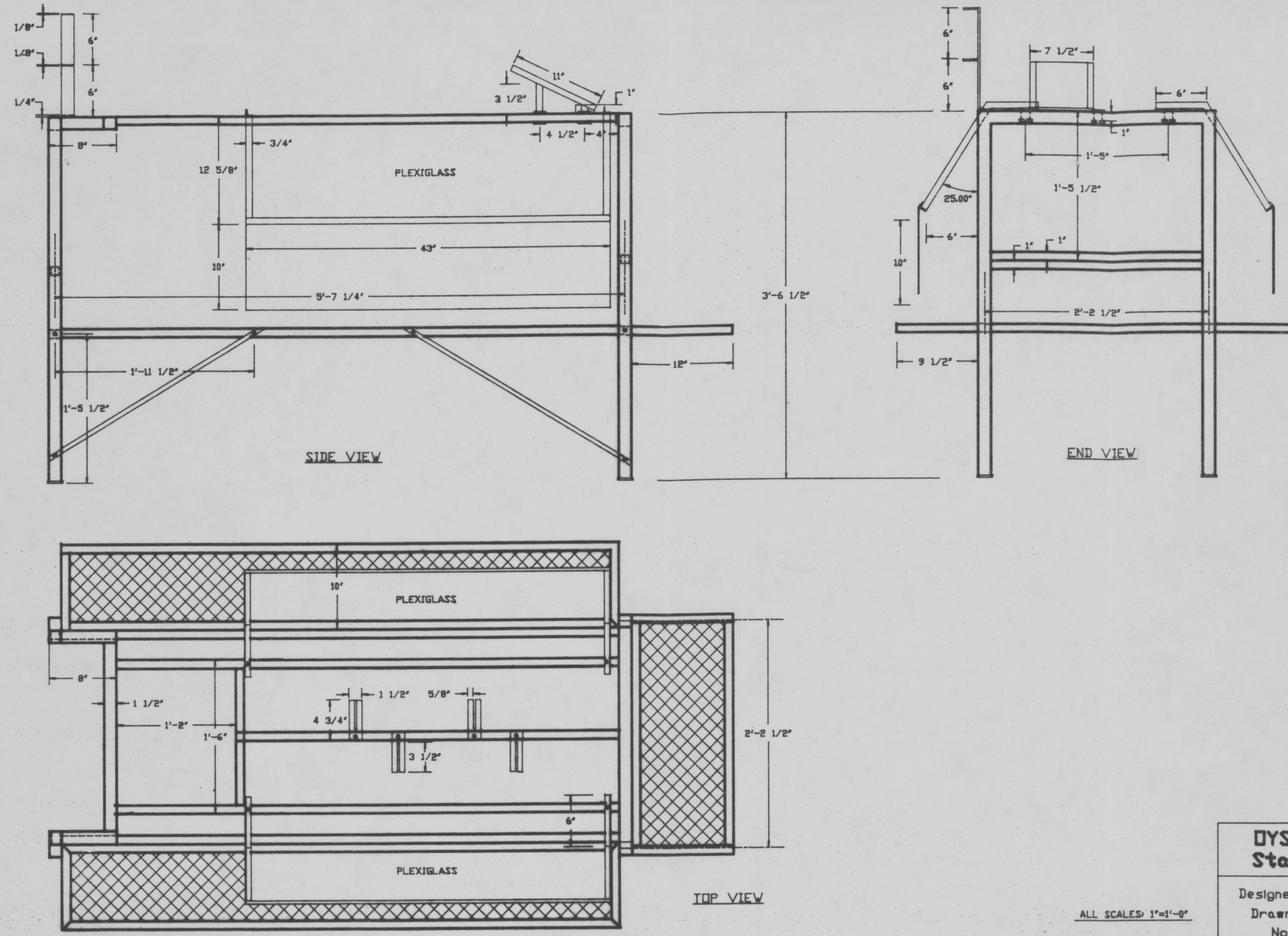
Wheaton, F. W., and A. G. Story, 1974. Oyster Severing Device: Mechanical Aspects. TRANSACTIONS of the ASAE, 17(5):934-936,941.

Wheaton, F. W., 1985. Freeze Shucking of Oysters. ASAE Paper No. 85-5007, St. Joseph, Michigan.

Wittaker, A. D., G. E. Miles, O. R. Mitchell, and L. D. Gaultney, 1987. Fruit Location in Partially Occluded Image. TRANSACTIONS of the ASAE, 30(3):591-600.

Zuech, N., 1988. Applying Machine Vision. John Wiley & Sons, New York, 1. Ed.

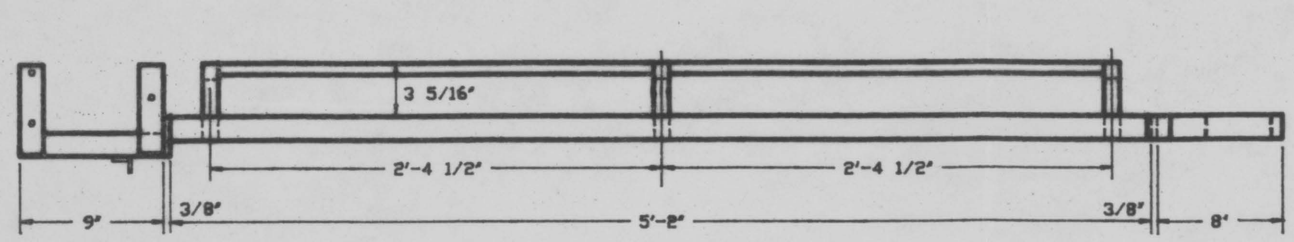
Appendix A. Drawings



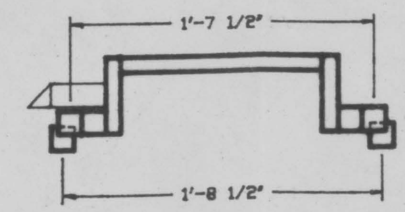
ALL SCALES: 1"=1'-0"

OYSTER SORTER
Stand Assembly

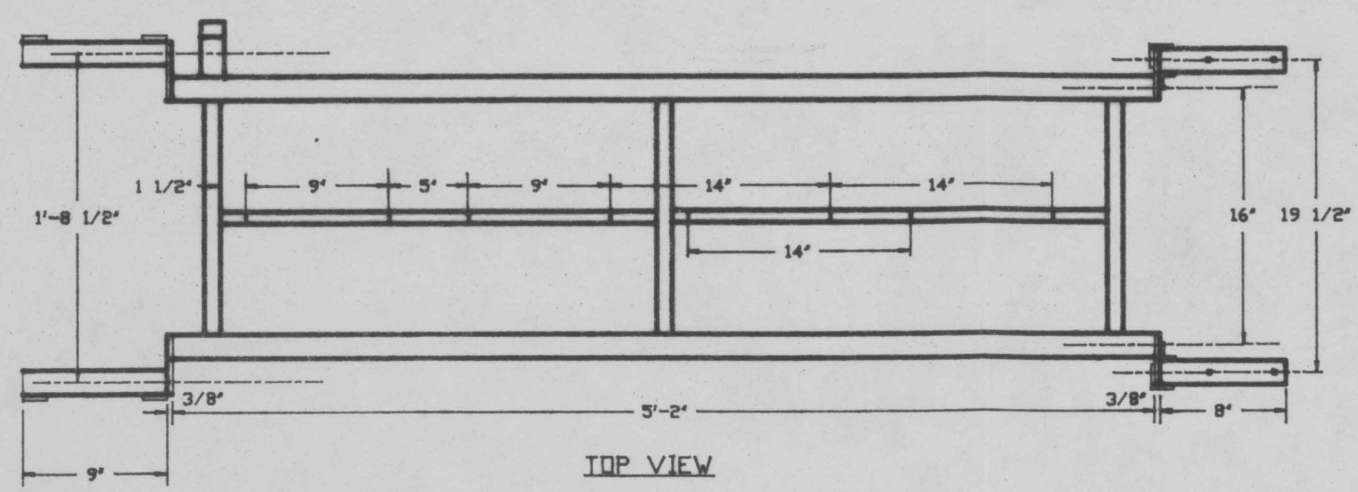
Designed by: Maria Koslav
Drawn by: Jack Davis
November, 1989



SIDE VIEW



END VIEW



TOP VIEW

ALL SCALES: 1"=1'-0"

OYSTER SORTER
Conveyor Belt Frame

Designed by: Maria Koslav
Drawn by: Jack Davis
November, 1989

Appendix B. Machine Components

Conveyor Components

[x] see documentation in Appendix A. Technical Drawings

Pos. No.	Units	Partname & Description	Catalog No.	Manufacturer
1[x]		Conveyor bed frame	C1018	Roanoke Steel Service Roanoke, VA 24016
	14'	1.5"x1.5"x3/16" sq. tubing		
	4'	1"x1"x1/8" sq. tubing		
	5'	3/4"x3/4"x1/8" sq. tubing		
	3'	2"x3/8" flat steel		
2	1	Skid plate stainless steel 30" x 60", 16 gauge	304SS	Edgcomb Metals Troy, VA 22974
3	1	Conveyor belt dark green, PU FDA approved, 13'long, 14"wide	E6/1UO/U	Siegling Belting Inglewood, NY
4	2	Take-up frame 3" adjustment 8 1/2" x 4 1/18"	100-3SF16	Browning Manufacturing Division Maysville, KY 41056
5	2	Take-up ball bearing 1" bore	100-MTWS216	Browning

6	2	Pillowblock ball bearing 1" bore	100-MPS-216	Browning
7	2	Conveyor pulley 1/4" rubber lagging, crowned, 4.5" dia, 16" wide, 1" bore	100-D4C-16X1/4	Browning
8	2	Gear belt pulley 1/2" face, 30 grooves, 3.581 pitch dia., 3/8" pitch	100-30LH050	Browning
9	1	Gear belt light duty, 1/2" wide 255 grooves, 3/8 pitch	100-255L050	Browning
10	1	Split taper bushing 1" bore	100-H1	Browning
11	1	Split taper bushing 5/8" bore	100-H5/8	Browning
12	1	Gear motor 90 VDC, 1/8 hp, 1.5 A max. 100 RPM	4Z382	Dayton Electric Manufacturing Co. Chicago, IL 60648
13*	 5"	Motor holder 4"x1" flat steel	C1018	Roanoke Steel Service
14	1	Motor speed controller 0 - 100 RPM	4Z829	Dayton Electric Mfg., Co.
15*	 35" 37"	Motor controller stand 1"x1"x1/8" angle iron 1"x1/8" flat steel	C1018	Roanoke Steel Service
16*	 19' 20' 10'	Conveyor stand 1.5"x1.5"x3/16" sq. tubing 1"x1"x1/8" sq. tubing 1"x1/4" flat steel	C1018	Roanoke Steel Service
17	4	Swivel casters 3" dia wheels, 3 5/8" high, 210 lbs max load	4X696	Graingers Roanoke, VA 24016
18	1	Casters brake kit	4X698	Graingers

19[x]		Container shelf	C1018	Roanoke Steel Service
	48'	1"x1"x1/8" angle iron		
	11'	8" wide steel mesh		
	2.5'	11" wide steel mesh		
20[x]		Splash guard	C1018	Roanoke Steel Service
	14'	3/4"x3/4"x1/8" angle iron		
	2	14"x42.5"x1/4" plexiglas sheet		
	2	11"x42.5" plastic curtain		
21[x]		Pressure regulator stand	C1018	Roanoke Steel Service
	10"	3"x3/16" flat steel		
	6"	3"x1/8" flat steel		
22[x]		Valve and nozzle holder	C1018	Roanoke Steel Service
	20"	1.75"x3/16" flat steel		
	40"	0.5" dia. steel rod		

Image System Components

Pos. No.	Units	Partname & Description	Catalog No.	Manufacturer
23	1	Television camera monochrome, 60 Hz, 9W, 120 VAC 16 mm lens	WV-1410	Panasonic Industrial Company Secaucus, NY 07094
24	1	Digitizer/ Transmitter	ImageWise	Micromint Inc. Vernon, CT 06066
25	1	Photographic flash light 3 flashes/s at ...	AUTO 360PX	Minolta Camera Co. Ltd. Japan
26	1	AC flash adapter	AC A 4	Minolta Camera Co. Ltd.
27	1	Phototransistor Peak response wavelength 800 nm max. power dissipation 200 mW	FPT110	Fairchild Inc. Mountain View, CA 94042
28	1	Infrared light emitting diode (IRED) Peak wavelength 940 nm Forward current 100 mA Power 170 mW	1N6266	GE Somerville, NJ
29	 12" 10" 10"	Transistor/IRED holder 1.5"x1.5x3/16" angle aluminum 1"x3/16" flat aluminum 1/2"x1/2" bar aluminum	6068	Edgcomb Metals
30	1	Comparator circuit for phototransistor signal		
31	1	Polarizer filter 40.5 mm dia.	PL	Quantaray Enterprises Beltsville, MD 20705
32	1	Polarizing sheet material 7.5" x 7.5" x 0.01"	D37/350	Edmund Scientific Co. Barrington, NJ 08007

33		Illumination chamber 16" x 13.5", 30" high		Roanoke Steel Service
	14 ft^2	sheet metal		
	7 ft	3/4"x3/4"x1/8" angle iron		
34	1	Flash light holder flat steel, 3/8" steel rod	C1018	Roanoke Steel Service
35	1	Camera holder		Minolta Camera Co. Ltd.
36	1	Personal Computer	JE1017	Jameco Electronics Inc. Belmont, CA 94002

Sorting System Components

Pos. No.	Units	Partname & Description	Catalog No.	Manufacturer
37	1	Water pressure regulator with pressure gauge 0 - 250 psi	11438-251	Spraying Systems Co. Wheaton, IL 60188
38	4	Spray valve Autojet, 3 cycles/s, 40 bar max. capacity 7.6 l/min	AA22AUH	Spraying Systems Co.
40	8	Nozzle tips Unijet, 25° spray angle, 0.17 GPM at 30 psi, brass	TP2502	Spraying Systems Co.
41	4	Nozzle caps Unijet, brass	CP1325	Spraying Systems Co.
42	4	Swivel body Unijet, brass	6240-1/4TT	Spraying Systems Co.
43	1	Air pressure regulator with pressure gauge 0 - 250 psi	04E11A13F	Parker Hannifin Corp. Applied Fluidpower Chesterfield, VA 23832
44	8	Fitting Prestolok, 1/8" NPT for 1/8" O. D. tubing	68PL-2-2	Parker Hannifin Corp.
45	7	Fitting Prestolok, 1/8" NPT for 1/4" O. D. tubing	68PL-4-2	Parker Hannifin Corp.
46	3	T-fitting Prestolok for 1/4" O. D. tubing	164PL-4	Parker Hannifin Corp.
47	25 ft	Tubing Polyethylen, 1/4" O. D.	E-43-250	Parker Hannifin Corp.

48	25 ft	Tubing Nylon, 1/8" O. D.	NN2-016	Parker Hannifin Corp.
49	4	Air manifold valve 120 VAC, 3 way normally closed	M31E1	Humphrey Products Kalamazoo, MI 49003
50	1	Air manifold aluminum, 4 stations	MM4	Humphrey Products
51	4	Solid state relay 110 - 240 VAC output 3 - 32 VDC input, 10 A	S505-OSJ410-000	Continental Industries, Inc. Mesa, AZ 82205
52	1	Power supply +/- 12 VDC output		
53	1	Inductive proximity switch	651810 010	Veeder-Root Inc. Hartford, CT 06102
54	1	Timing wheel 9" dia., 15 pulses/rotation		
55	1	Microcontroller 24 programmable I/O lines 2 programmable counter/timers EPROM stored software	VItrax IX	Sintec Co. Frenchtown, NJ 08825

Appendix C. Software

```
PROGRAM Sort(input,output,picfile);

{ Program to take and analyze pictures of oyster meats   }
{ for size grading purposes, Sept. 1989  Maria Koslav    }


{ Uses Image Acquisition Routines of ImageWise Software }
{ written by Ciarcia's Circuit Cellar, 1987             }
{ Copyright (c) 1987, Ciarcia's Circuit Cellar          }
{    All Rights Reserved                                }

{$U- control-break checking during execution            }
{$C- control-break checking during I/O operations       }
{$R- array range checking                               }

{-------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar          }
{    All Rights Reserved                                }

Const

{--- defaults for serial port setup                     }
 COMport   = $3F8;               { 3F8 = COM1           }

{--- resolution control bytes to receiver & transmitter }
 fullres   = $80;                { set high resolution  }
 halfres   = $81;                { set high resolution  }
 quartres  = $82;                { set high resolution  }

{--- control bytes from transmitter                     }
 fieldsync = $40;                { new field!           }
 linesync  = $41;                { new line             }
 fldend    = $42;                { end of field         }
 rep1      = $80;                { repeat x1            }
 rep16     = $90;                { repeat x16           }

{--- image structure                                    }
 maxbit    = $3F;                { bits used in pel     }
 maxpel    = 255;                { highest pel index    }
 maxline   = 243;                { highest line index   }
 maxbuffer = 32766;              { highest "INT" index  }

 maxtime   = 100;                       { serial timeout }
```

```
{--- Async card constants                                  }
 XON       = $11;
 XOFF      = $13;

 DataReady = $01;              { receive data ready    }
 THRE      = $20;              { transmit data ready   }

{-----------------------------------------------------------}
       (* Maria Koslav, 1989 *)
const
      bitsec     = 288;     { default serial speed x 100 }
      Com2Data   = $2f8;    { 2F8 = COM2            }
      Com2LSR    = $2fd;

Type  Data       = String[6];
{-----------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar             }
{    All Rights Reserved                                   }

TYPE
 bitrng    = 0..maxbit;          { bit range               }
 pelrng    = 0..maxpel;          { pel indexes             }
 linerng   = 0..maxline;         { line indexes            }
 subrng    = 0..maxbuffer;       { raw data indexes        }

 pelrec    = RECORD              { one scan line           }
           syncL : BYTE;
           pels  : ARRAY[pelrng] OF BYTE;
           END;

 linerec   = RECORD              { complete binary field }
           syncF : BYTE;
           lines : ARRAY[linerng] OF pelrec;
           syncE : BYTE;
           END;

 rawrec    = ARRAY[subrng] OF INTEGER;

 picptr    = ^pictype;                  { picture ptr    }
 pictype   = RECORD CASE INTEGER OF     { picture formats}
           0 : (fmt : linerec);
           1 : (words : rawrec);
           1 : (bytes : BYTE);          { dummy          }
           END;

 histtype  = ARRAY[bitrng] OF REAL;     { pel histograms }

 regrec = RECORD CASE INTEGER OF
         1 : (AX : INTEGER;
              BX : INTEGER;
              CX : INTEGER;
              DX : INTEGER;
              BP : INTEGER;
              SI : INTEGER;
              DI : INTEGER;
              DS : INTEGER;
              ES : INTEGER;
              FLAGS : INTEGER);
         2 : (AL,AH : BYTE;
              BL,BH : BYTE;
              CL,CH : BYTE;
              DL,DH : BYTE);
```

```
        END;

 byteptr   = ^BYTE;                        { general ptr    }
 strtype   = STRING[255];                  { strings        }
 Hextype    = STRING[4];          { input & output string }

{------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar         }
{    All Rights Reserved                               }

VAR

 COMerror  : INTEGER;

{ Async card global variables                          }
 comdata : INTEGER;              { data for async I/O    }
 comien  : INTEGER;              { interrrupt enable reg }
 comiir  : INTEGER;              { interrupt ID reg      }
 comlcr  : INTEGER;              { line control reg      }
 commcr  : INTEGER;              { modem control reg     }
 comlsr  : INTEGER;              { line status reg       }
 commsr  : INTEGER;              { modem status reg      }

{ Picture pointer                                      }
 picfile   : FILE OF pictype;
 pic0      : picptr;
 pic1      : picptr;
 pic2      : picptr;
 pic3      : picptr;
 histo     : histtype;

 IOerror   : BYTE;
 filespec  : strtype;
 filespec2 : strtype;
{------------------------------------------------------}
      (* Maria Koslav, 1989 *)

VAR
 manual       : BOOLEAN;              { FALSE when in auto    }
 more         : CHAR;                 { answers to question   }
 TempFName    : Strtype;              { Temporary file        }
 DetLine, Level                                : integer;
 LCol, RCol                                    : integer;
 PCtOyster                                     : Array[1..3] of REal;
 SizeR, SizeL                                  : String[1];
 Q, LL, No_Oyster                              : integer;
 StaL, StaR, SeL, SeR, CoL, CoR, TotalCT       : integer;
 AvR, PDevL, PDevR, PDevRef                    : REAL;
 WeightArea                                    : integer;
 AreaLeft, Arearight, Weightleft, Weightright : Real;

 Data_File                                     : String[20];
 OutputFile                                    : Text;
 Data1, Data2, Data3, Data4, Data5, DAta6      :  String[12];

    Strg        : String[6];
    noz_dat     : array[1..5] of string[3];
{------------------------------------------------------}
{------------------------------------------------------}
{------------------------------------------------------}
{------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar         }
```

```
{    All Rights Reserved                                }
{ Set up the async card                                  }
{ Rate is in bits/second                                 }

PROCEDURE ComOn(rate : INTEGER);

CONST
 serialmax = 1152.0;          { rate -> divisors      }

VAR
 dummy      : BYTE;
 counts     : INTEGER;

BEGIN

{--- set up global variables                             }

 comdata := comport;
 comien  := comport + 1;
 comiir  := comport + 2;
 comlcr  := comport + 3;
 commcr  := comport + 4;
 comlsr  := comport + 5;
 commsr  := comport + 6;

{--- set up port registers                               }

 counts := Trunc(serialmax/rate);

 Port[comlcr]  := $80;          { set DLAB to set rate  }
 Port[comdata] := Lo(counts);   { set divisor LSB       }
 Port[comien]  := Hi(counts);   { set divisor MSB       }
 Port[comlcr]  := $13;          {   no pty 1 stop 8 dat }

 dummy := Port[comdata];        { discard pending char  }

END;

{-------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar          }
{    All Rights Reserved                                }
{ Send a byte to the serial port                        }
{ If there's an XOFF in the receiver, we wait...        }
{ This is not likely, but it has been known to happen   }

PROCEDURE SendByte(databyte : BYTE);

BEGIN

 COMerror := 0;                 { can't have error...   }

 WHILE (Port[comlsr] AND THRE) = 0 DO;     { send done?  }

 WHILE Port[comdata] = XOFF DO; { XOFF pending?         }

 Port[comdata] := databyte;     { send data             }

END;

{-------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar          }
{    All Rights Reserved                                }
```

```
{ converts one 4-bit nybble to a character                    }

FUNCTION NybToHex(nybval:INTEGER) : HexType;

BEGIN
 CASE nybval OF
  0..9   : NybToHex := Chr($30+nybval);
  10..15 : NybToHex := Chr($41+nybval-10);
 END;  { CASE }
END;

{-------------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar                }
{    All Rights Reserved                                      }
{ converts byte to characters                                 }

FUNCTION ByteToHex(intval:INTEGER) : Hextype;

VAR
 tempstring     : Hextype;

BEGIN

 tempstring := '00';

 tempstring[1] := NybToHex((intval AND $00F0) DIV 16);
 tempstring[2] := NybToHex( intval AND $000F);

 ByteToHex := tempstring;

END;

{-------------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar                }
{    All Rights Reserved                                      }
{ Present message, return boolean response                    }

FUNCTION Askit(msg : strtype) : BOOLEAN;

VAR
 resp      : STRING[5];

BEGIN
 Write(msg,' ');                  { present question          }
 Readln(resp);                    { get some answer           }

 Askit := FALSE;
 IF Length(resp) <> 0             { categorize response       }
  THEN IF UpCase(resp[1]) = 'Y'
        THEN Askit := TRUE;

END;

{-------------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar                }
{    All Rights Reserved                                      }
{ Allocate and initialize the picture buffer                  }

PROCEDURE PicSetup(VAR newpic : picptr);

VAR
 pels      : pelrng;
```

```
 lines       : linerng;

BEGIN

 IF newpic <> NIL               { discard if allocated   }
  THEN Dispose(newpic);

 New(newpic);                   { allocate new array     }

END;

{-------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar          }
{    All Rights Reserved                                }
{ Set up frame and line syncs in a buffer               }
{ This should be done only in freshly allocated buffers }

PROCEDURE SetSyncs(pic1 : picptr);

VAR
 lndx       : linerng;          { index into lines       }

BEGIN

 pic1^.fmt.syncF := fieldsync;  { set up empty picture   }

 FOR lndx := 0 TO maxline DO BEGIN
  pic1^.fmt.lines[lndx].syncL := linesync;
  FillChar(pic1^.fmt.lines[lndx].pels[0],maxpel+1,0);
 END;

 pic1^.fmt.syncE := fldend;     { set ending control     }

END;

{-------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar          }
{    All Rights Reserved                                }
{ Get a picture from the transmitter                    }
{ The bit rate depends on which PC you're using...      }
{ An 8 MHz AT can handle 28.8 K bits/sec                }
{ Sets RTS and DTR to switch the relay box before       }
{  taking the picture, restores normal display after    }
{ Some debugging statements are commented out... you    }
{  may need them to get your system running             }

{ Getpicture changed to get a picture after the         }
{ occurrence of a trigger signal at port[$3fe], bit 6   }
{ Flash trigger at port[ComMCR], bit 1.                 }
{     August 25, 1989  -- Maria Koslav                  }

       (* Giarcia' s Circuit Cellar  *)

PROCEDURE GetPicture(pic : picptr;
                     resol : BYTE);

VAR
 picbyte    : BYTE;             { byte from transmitter }
 bptr       : byteptr;          { fake pointer to pic   }
 K, I       : Integer;
 LED        : BYTE;
```

```
BEGIN

 K := 0;

 bptr := Ptr(Seg(pic^),Ofs(pic^)-1);  { preset for loop }

 SendByte(resol);                 { specify resolution    }

TextColor(white);
Gotoxy(45,24);write('Waiting for LED signal');
repeat
LED := Port[$3fe];
until ((LED and 64) = 0);
Gotoxy(45,24);write('                         ');

 Port[comMCR] := $02;             { PC <-> trans serial   }
                                  { camera -> monitor     }
 SendByte(XON);                   { prompt transmitter    }

 REPEAT                           { for each line         }
    bptr := Ptr(Seg(bptr^),Ofs(bptr^)+1);   { tick ptr     }
    WHILE ((Port[comLSR] AND DataReady) = 0) AND
          NOT KeyPressed DO;        { stall waiting          }
    Port[comMCR] := $03;            { falsh on               }
    bptr^ := Port[comdata];         { snag the byte          }
 UNTIL (bptr^ = fldend) OR KeyPressed;

 Port[comMCR] := $00;

END;

{------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar         }
{    All Rights Reserved                               }
{ Decompress pic1 into pic2                            }

PROCEDURE Expand(pic1,pic2 : picptr);

CONST
 errthresh = 10;          { max errors in frame        }

VAR
 bptr      : ^byte;
 lndx      : linerng;
 pndx      : pelrng;
 overflow  : BOOLEAN;
 oldbyte   : BYTE;
 reps      : INTEGER;
 frametop  : BOOLEAN;
 giveup    : BOOLEAN;
 errcount  : INTEGER;

BEGIN

 bptr := Ptr(Seg(pic1^),Ofs(pic1^));

 SetSyncs(pic2);                  { fill in the syncs     }

 lndx := 0;
 pndx := 0;

 frametop := TRUE;
```

```
 giveup := FALSE;
 errcount := 0;
 WHILE (bptr^ <> fldend) AND NOT giveup
 DO BEGIN { and now the data...    }
  CASE bptr^ OF
   fieldsync : BEGIN
              IF (lndx <> 0) OR (pndx <> 0)
               THEN BEGIN
                Writeln('Field sync found after data');
               END;
              oldbyte := 0;
              frametop := TRUE;
(*            Writeln('Field sync'); *)
             END;
   linesync : BEGIN
              IF (lndx < maxline) AND NOT frametop
               THEN lndx := lndx + 1
               ELSE frametop := false;
              oldbyte := 0;
              pndx := 0;
              overflow := FALSE;
(*            Write('.'); *)
             END;
   fldend : BEGIN                { can't get here...     }
             Writeln;
             Writeln('Surprise at having found field end!');
            END;
   ELSE BEGIN
    CASE (bptr^ AND $F0) OF
     $00..$3F : BEGIN
                pic2^.fmt.lines[lndx].pels[pndx] := bptr^;
                oldbyte := bptr^;
                IF pndx < maxpel
                 THEN BEGIN
                  pndx := pndx + 1;
                  IF overflow
                   THEN BEGIN
                    Write('Too much data on line ',lndx:3);
                    Writeln('  pel data ',ByteToHex(bptr^));
                    errcount := Succ(errcount);
                   END;
                 END
                 ELSE BEGIN
                  pndx := 0;
                  overflow := TRUE;
                 END;
(*              Writeln('Data: ',ByteToHex(bptr^)); *)
               END;
     repl   : BEGIN
                FOR reps := 1 TO (bptr^ AND $0F) DO BEGIN
                 pic2^.fmt.lines[lndx].pels[pndx] := oldbyte;
                 IF pndx < maxpel
                  THEN BEGIN
                   pndx := pndx + 1;
                   IF overflow
                    THEN BEGIN
                     Write('Too much data on line ',lndx:3);
                     Writeln('   1x rep ',ByteToHex(bptr^));
                     errcount := Succ(errcount);
(*                   pndx := 0; *)
                    END
                  END
```

```
                    ELSE BEGIN
                     pndx := 0;
                     overflow := TRUE;
                     END;
(*                 Writeln('Rep1: ',ByteToHex(bptr^)); *)
                  END;
                 END;
      rep16 : BEGIN
                  FOR reps := 1 TO (16 * (bptr^ AND $0F)) DO BEGIN
                   pic2^.fmt.lines[lndx].pels[pndx] := oldbyte;
                   IF pndx < maxpel
                    THEN BEGIN
                     pndx := pndx + 1;
                     IF overflow
                      THEN BEGIN
                       Write('Too much data on line ',lndx:3);
                       Writeln('  16x rep ',ByteToHex(bptr^));
                       errcount := Succ(errcount);
(*                     pndx := 0; *)
                      END
                    END
                    ELSE BEGIN
                     pndx := 0;
                     overflow := TRUE;
                    END;
(*                 Writeln('Rep16: ',ByteToHex(bptr^)); *)
                  END;
                 END;
      ELSE BEGIN
       Writeln('Garbage byte: ',ByteToHex(bptr^),
                   ' at line ',lndx,' pel ',pndx);
       errcount := Succ(errcount);
      END;
     END;
    END;
   END;
   IF errcount > errthresh
    THEN giveup := TRUE;
   bptr := Ptr(Seg(bptr^),Ofs(bptr^)+1);   { next input byte }
  END;

  IF giveup
   THEN BEGIN
    Writeln('Too many errors -- giving up!');
   END;

(* Writeln; *)

END;

{-------------------------------------------------------}
{ Copyright (c) 1987, Ciarcia's Circuit Cellar          }
{     All Rights Reserved                               }
{ Save picture file on disk                             }
{ Uses the smallest number of blocks to fit the data    }

{ SavePicture changed to close the file each time       }
{     November 18, 1988 -- R.K. Byler                   }

PROCEDURE SavePicture(filespec : strtype;
                      pic : picptr);
VAR
```

```
 ndx       : subrng;              { index into word array }
 rndx      : REAL;                {  real equivalent      }
 nblocks   : INTEGER;             { number of disk blocks }
 xfered    : INTEGER;             { number actually done  }

 pfile     : FILE;                { untyped file for I/O  }

BEGIN

 Writeln('Writing ',filespec);
 Assign(pfile,filespec);
 Rewrite(pfile);

 ndx := 0;                        { start with first word }

 Write('  Data length = ');
 WHILE (ndx < maxbuffer) AND      { WHILE not end of pic  }
       (Lo(pic^.words[ndx]) <> fldend) AND
       (Hi(pic^.words[ndx]) <> fldend) DO
   ndx := ndx + 1;

 ndx := ndx + 1;                  { fix 0 origin          }

 rndx := 2.0 * ndx;               { allow >32K numbers... }
 Write(rndx:6:0,' bytes, file length = ');

 nblocks := ndx DIV 64;           { 64 words = 128 bytes  }

 IF (ndx MOD 64) <> 0             { partial block?        }
  THEN nblocks := nblocks + 1;

 rndx := 128.0 * nblocks;         { actual file size      }
 Writeln(rndx:6:0,' bytes');

 BlockWrite(pfile,pic^.words[0],nblocks,xfered);

 IF xfered <> nblocks             { completed?            }
  THEN BEGIN
   Writeln('Problem writing the file, error code: ',
            IOerror);
   Writeln('  Blocks computed: ',nblocks);
   Writeln('  Blocks written:  ',xfered);
  END;
 CLOSE(pfile)
END;
{-------------------------------------------------------}
{-------------------------------------------------------}
{-------------------------------------------------------}
{-------------------------------------------------------}

(*******************************************************)
       (* Maria Koslav, 1989 *)

Procedure SendStrng(Strng:data);

var x : Char;
    D : integer;

begin
  for D := 1 to Length(strng) do
    begin
     x := Strng[D];
```

```
     delay(6);    {allow time for Vitrax to process data, high baud}
     repeat until (Port[Com2LSR] and 32) = 32;{ for low baud rates }
     Port[Com2data] := ord(x);
    end;
end;

(****************************************************)
       (* Maria Koslav, 1989 *)

Procedure DOyster(pic2 : picptr);

Label              CheckAnotherLine, NoOysterFound;

var lndx                                  : linerng;
    pndx                                  : pelrng;
    PelCt, SumNeigh, K, NN                : Integer;
    StartPel, StartLine                   : Integer;
    LimitCol, DetectLine                  : integer;
    Dummy                                 : Real;

begin

No_oyster := 0;
LCol := 30;
Rcol := 255;

For LL := 3 downto 1 do
begin

Detectline := DetLine; NN := 0;

CheckAnotherLine:

   Case LL of
     3 : begin pndx := 30; LimitCol := 105; Level := 27; end;
     2 : begin pndx := 105; LimitCol := 160; level := 20; end;
     1 : begin pndx := 160; LimitCol := 250; level := 2; end;
   end;


  repeat
  PelCt := 0;
      Repeat
         pndx := pndx + 5;
      Until ((pic2^.fmt.lines[Detectline].pels[pndx] >= Level) or
(pndx>=LimitCol));
      Repeat
         PelCT := PelCt + 1;
         pndx := pndx + 1;
      until ((pic2^.fmt.lines[Detectline].pels[pndx] < Level) or (pndx
>=LimitCol));
  until (pelCt >= 7) or (pndx >= LimitCol);

If (pndx >= LimitCol) and (NN <= 75) then
   begin
      Detectline := Detline - 35;
      NN := NN + 15;
      Detectline := Detectline + NN;
      goto CheckAnotherLine;
   end;

If pndx >= LimitCol THEN begin
```

```
            {filespec := 'xxx';
            SavePicture(filespec,pic1);} { save it away  pic1 -
compressed }
            No_Oyster := 1; goto NoOysterFound; end;

pndx := pndx-1;
lndx := DetectLine;
StartPel := pndx;
StartLine := lndx;
PctOyster[LL] := 0;

K := 2;

Repeat
SumNeigh := 0;
 CASE K of
 0..1 : begin
  IF pic2^.fmt.lines[lndx].pels[pndx+1] >= Level THEN SumNeigh :=
SumNeigh+4;
  IF pic2^.fmt.lines[lndx-1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+8;
  IF pic2^.fmt.lines[lndx-1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+16;
  IF pic2^.fmt.lines[lndx-1].pels[pndx-1]>=Level THEN SumNeigh :=
SumNeigh+32;
  IF pic2^.fmt.lines[lndx].pels[pndx-1] >= Level THEN SumNeigh :=
SumNeigh+64;
  IF pic2^.fmt.lines[lndx+1].pels[pndx-1]>=Level THEN SumNeigh :=
SumNeigh+128;
  IF pic2^.fmt.lines[lndx+1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+1;
  IF pic2^.fmt.lines[lndx+1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+2;
  K := 6;
        end;
 2..3 : begin
  IF pic2^.fmt.lines[lndx].pels[pndx+1] >= Level THEN SumNeigh :=
SumNeigh+1;
  IF pic2^.fmt.lines[lndx-1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+2;
  IF pic2^.fmt.lines[lndx-1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+4;
  IF pic2^.fmt.lines[lndx-1].pels[pndx-1]>=Level THEN SumNeigh :=
SumNeigh+8;
  IF pic2^.fmt.lines[lndx].pels[pndx-1] >= Level THEN SumNeigh :=
SumNeigh+16;
  IF pic2^.fmt.lines[lndx+1].pels[pndx-1]>=Level THEN SumNeigh :=
SumNeigh+32;
  IF pic2^.fmt.lines[lndx+1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+64;
  IF pic2^.fmt.lines[lndx+1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+128;
  K := 0;
        end;
 4..5 : begin
  IF pic2^.fmt.lines[lndx].pels[pndx+1] >= Level THEN SumNeigh :=
SumNeigh+64;
  IF pic2^.fmt.lines[lndx-1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+128;
  IF pic2^.fmt.lines[lndx-1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+1;
  IF pic2^.fmt.lines[lndx-1].pels[pndx-1]>=Level THEN SumNeigh :=
```

```
SumNeigh+2;
  IF pic2^.fmt.lines[lndx].pels[pndx-1] >= Level THEN SumNeigh :=
SumNeigh+4;
  IF pic2^.fmt.lines[lndx+1].pels[pndx-1]>=Level THEN SumNeigh :=
SumNeigh+8;
  IF pic2^.fmt.lines[lndx+1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+16;
  IF pic2^.fmt.lines[lndx+1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+32;
  K := 2;
        end;
 6..7 : begin
  IF pic2^.fmt.lines[lndx].pels[pndx+1] >= Level THEN SumNeigh :=
SumNeigh+16;
  IF pic2^.fmt.lines[lndx-1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+32;
  IF pic2^.fmt.lines[lndx-1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+64;
  IF pic2^.fmt.lines[lndx-1].pels[pndx-1]>=Level THEN SumNeigh :=
SumNeigh+128;
  IF pic2^.fmt.lines[lndx].pels[pndx-1] >= Level THEN SumNeigh :=
SumNeigh+1;
  IF pic2^.fmt.lines[lndx+1].pels[pndx-1]>=Level THEN SumNeigh :=
SumNeigh+2;
  IF pic2^.fmt.lines[lndx+1].pels[pndx] >= Level THEN SumNeigh :=
SumNeigh+4;
  IF pic2^.fmt.lines[lndx+1].pels[pndx+1]>=Level THEN SumNeigh :=
SumNeigh+8;
  K := 4;
       end;
end;

      repeat
        SumNeigh := SumNeigh shr 1;
        K := K+1;
        If K = 8 then K := 0;
      until ((SumNeigh mod 2) <> 0);

      CASE K of
      0 : begin  Lndx  := lndx;   Pndx  := pndx+1;  end;
      1 : begin  Lndx  := lndx-1; Pndx  := pndx+1;  end;
      2 : begin  Lndx  := lndx-1; Pndx  := pndx;    end;
      3 : begin  Lndx  := lndx-1; Pndx  := pndx-1;  end;
      4 : begin  Lndx  := lndx;   Pndx  := pndx-1;  end;
      5 : begin  Lndx  := lndx+1; Pndx  := pndx-1;  end;
      6 : begin  Lndx  := lndx+1; Pndx  := pndx;    end;
      7 : begin  Lndx  := lndx+1; Pndx  := pndx+1;  end;
      end;


If ((pic2^.fmt.lines[lndx].pels[pndx+1] < Level) Xor
    (pic2^.fmt.lines[lndx].pels[pndx-1] < Level)) THEN
  If (K < 4) THEN Pctoyster[LL] := pctoyster[LL] + pndx
             ELSE Pctoyster[LL] := pctoyster[LL] - (pndx-1);

If ((pic2^.fmt.lines[lndx].pels[pndx+1] < Level) and
    (pic2^.fmt.lines[lndx].pels[pndx-1] < Level)) THEN
             PCtOyster[LL] := PCtOyster[LL] + 1;

until ((pndx = StartPel) and (lndx = StartLine));
```

```
end;

NoOysterFound:

Dummy := PCtoyster[1] * (PDevL/PdevRef)*((AvR-PCtoyster[2])/AvR);
PCtOyster[1] := PCtoyster[1] + Dummy;
Dummy := PCtoyster[3] * (PDevR/PdevRef)*((AvR-PCtoyster[2])/AvR);
PCtOyster[3] := PCtoyster[3] + Dummy;

  Case WeightArea of
2 : begin   Arealeft := PCtOyster[1] * 1.09053;
            Arearight := PCtOyster[3] * 1.09864;
            SizeL := '0';  SizeR := '0';
               If AreaLeft < 1341.89 then  SizeL := '2';     {mm square}
               If AreaLeft > 2032.26 then  SizeL := '4';     {mm square}
               If AreaRight < 1341.89 then  SizeR := '1';   {mm square}
               If AreaRight > 2032.26 then  SizeR := '3'; end; {mm
square}
1 : begin   Weightleft :=  - 1.52754 + PCtOyster[1] * 0.009582;
            Weightright := PCtOyster[3] * 0.011404 - 2.5153;
            SizeL := '0';  SizeR := '0';
               If WeightLeft < 7.62979 then  SizeL := '2';   {gram}
               If WeightLeft > 13.1038 then  SizeL := '4';   {gram}
               If WeightRight < 7.62979 then  SizeR := '1';  {gram}
               If WeightRight > 13.1038 then  SizeR := '3';  end;
{gram}
  end;

If No_Oyster = 1 then begin
    If LL = 3 then SizeR := 'N';
    If (LL = 1) or (LL = 2) then SizeL := 'N';
    No_Oyster := 0;
  end;

    Strg := SizeL + SizeR + 'N' + #13;
    SendStrng(Strg);

end;

(********************************************************)
       (* Maria Koslav, 1989 *)

Procedure ScreenMask;

Var     a,b : real;

begin
 Textmode;
 ClrScr;
 gotoXY(20,1);  write('************************************');
 GotoXY(20,2);  write('*        OYSTER MEAT SORTER        * ');
 GotoXY(20,3);  write('*               1989               * ');
 gotoXY(20,4);  write('************************************');

TextColor(LightRed);
Gotoxy(45,7);          Write('Threshold level ');
Gotoxy(45,8);          Write('for pictures ?       ',level:3);
Gotoxy(45,10);         Write('Detector line ');
Gotoxy(45,11);         Write('in picture ?         ',DEtline:3);

TextColor(LightGreen);
gotoxy(14,7); Write('   Oyster Count ');
```

```
gotoxy(14,8); Write('for each Container ');

Gotoxy(12,10);           Write('-------------------------');
gotoxy(1,11); Write('STANDARDS  |            |            |');
Gotoxy(12,12);           Write('-------------------------');
gotoxy(1,13); Write('SELECTS    |            |            |');
Gotoxy(12,14);           Write('-------------------------');
gotoxy(1,15); Write('COUNTS     |            |            |');
Gotoxy(12,16);           Write('-------------------------');

Gotoxy(12,17);           Write('      ^           ^');
Gotoxy(12,18);           Write('      ^           ^');

Gotoxy(12,19);           Write('-------------------------
---------');
gotoxy(1,20); Write('Last Oyst. |            |            |
|       |');
Gotoxy(12,21);           Write('-------------------------
---------');
Gotoxy(50,18);Write('Total Number ');
Gotoxy(49,22);Write('Oysters Graded');

Gotoxy(12,22);           Write('      ^           ^');
Gotoxy(12,23);           Write('      ^  Camera   ^');

end;

(**************************************************************)
       (* Maria Koslav, 1989 *)

Procedure CountAndDisplay;

begin

TextColor(Lightcyan);

Case SizeL of
'2' :  begin  CoL := CoL + 1;  TotalCT := TotalCT + 1;
        gotoxy(15,15); Write(CoL:5); gotoxy(15,20); Write (' Count   ');
end;
'0' :  begin  StaL := StaL + 1;  TotalCT := TotalCT + 1;
        gotoxy(15,11); Write(StaL:5); gotoxy(15,20); Write('Standard ');
end;
'4' :  begin  SeL := SeL + 1;  TotalCT := TotalCT + 1;
        gotoxy(15,13); Write(SeL:5); gotoxy(15,20); Write (' Select
');end;
'N' :  begin gotoxy(14,20); Write('NO OYSTER');  end;
     end;

Case SizeR of
'1' :  begin  CoR := CoR + 1;  TotalCT := TotalCT + 1;
        gotoxy(27,15); Write(CoR:5); gotoxy(27,20); Write(' Count   ');
end;
'0' :  begin  StaR := StaR + 1;  TotalCT := TotalCT + 1;
        gotoxy(27,11); Write(StaR:5); gotoxy(27,20); Write('Standard ');
end;
'3' :  begin  SeR := SeR + 1;  TotalCT := TotalCT + 1;
        gotoxy(27,13); Write(SeR:5); gotoxy(27,20); Write(' Select  ');
end;
'N' :  begin gotoxy(26,20); Write('NO OYSTER');  end;

     end;
```

```
gotoxy(37,20); Write(PCtOyster[2]:6:0);
gotoxy(53,20); Write(TotalCT:5);

     Str(PCtOyster[2]:12:1,Data1);
     Str(PCtOyster[1]:12:1,Data2);
     Str(PCtOyster[3]:12:1,Data4);
     Str(TotalCT:5,Data6);
     Writeln(OutputFile,Data6,Data1,Data2,'       ',SizeL,'
',Data4,'      ',SizeR);

end;

(*******************************************************)
       (* Maria Koslav, 1989 *)

Procedure GrabPicture;

 begin
    IF (ParamStr(2) = '/n') OR (ParamStr(3) = '/n')
         THEN manual := FALSE
         ELSE manual := TRUE;
      ComOn(bitsec);                        { set up serial port     }

repeat
      TextColor(white);
      Sound(500); Delay(400); Nosound;
      GetPicture(pic1,fullres);              { get the byte           }
      gotoxy(45,24); Write('Expanding                 ');
      Expand(pic1,pic2);
      gotoxy(45,24); Write('                          ');

      DOyster(pic2);
      CountandDisplay;
until keypressed;

end;

(*******************************************************)
       (* Maria Koslav, 1989 *)

BEGIN   {    main    }

noz_dat[1]:='180';
noz_dat[2]:='255';
noz_dat[3]:='330';
noz_dat[4]:='400';
noz_dat[5]:='040';
SendStrng('1'+noz_dat[1]+'T'+#13);
delay(100);
SendStrng('2'+noz_dat[2]+'T'+#13);
delay(100);
SendStrng('3'+noz_dat[3]+'T'+#13);
delay(100);
SendStrng('4'+noz_dat[4]+'T'+#13);
delay(100);
SendStrng(noz_dat[5]+'L'+#13);

        Writeln('Output file name ? ');
        Readln(Data_File);
        Assign(OutputFile,Data_File);
```

```
        Rewrite(outputFile);
         Writeln(OutputFile,'Pixel Data');
         Writeln(OutputFile,'No.   Reference   Left     Size      Right
 Size');

               StaL := 0; StaR := 0;
               SeL := 0; SeR := 0;
               CoL := 0; CoR := 0;
               TotalCT := 0;

               pic1 := NIL;     { ensure new alloc      }
               PicSetup(pic1);       { set up picture array  }
               pic2 := NIL;
               PicSetUp(pic2);

  Writeln('This program sorts oysters ');
  Writeln;
  Write('Detector line ? ');
  readln(DetLine);
  Write('Threshold level ? ');
  readln(level);
  {Write('Average of Reference Target ? ');
  readln(AvR);
  Write('Percent SDev. of left object ? ');
  readln(PDevL);
  Write('Percent SDev. of right object ? ');
  readln(PDevR);
  Write('Percent SDev. of reference target ? ');
  readln(PDevRef);
}  writeln; Avr := 776.10; PDevL := 2.3; PDevR := 2.81; PDevRef := 3.59;

  Write('Weight sorting  =  1  ; Area sorting   =  2 ');
  readln(WeightArea);
  ScreenMask;
  GrabPicture;

END.

(**************************************************************)
(**************************************************************)
(**************************************************************)
```

Appendix D. Calibration Data

Uncorrected Pixel Data

Target Size		
	Left :	1871.0 [mm^2]
	Right :	1890.0 [mm^2]
	Center :	731.0 [mm^2]

No.	Target Position		
	Left	Center	Right
1	1766.0	816.0	1782.0
2	1674.0	753.0	1671.0
3	1686.0	763.0	1684.0
4	1756.0	808.0	1770.0
5	1661.0	745.0	1655.0
6	1659.0	747.0	1653.0
7	1668.0	750.0	1662.0
8	1673.0	754.0	1669.0
9	1675.0	762.0	1674.0
10	1743.0	809.0	1768.0
11	1752.0	805.0	1768.0
12	1662.0	743.0	1656.0
13	1666.0	748.0	1660.0
14	1719.0	791.0	1728.0
15	1754.0	810.0	1770.0
16	1721.0	795.0	1727.0
17	1716.0	785.0	1718.0
18	1653.0	743.0	1653.0
19	1652.0	740.0	1649.0
20	1720.0	792.0	1724.0
21	1657.0	746.0	1655.0

22	1754.0	807.0	1769.0
23	1657.0	738.0	1648.0
24	1731.0	805.0	1745.0
25	1718.0	797.0	1733.0
26	1671.0	759.0	1674.0
27	1759.0	813.0	1776.0
28	1736.0	805.0	1748.0
29	1719.0	796.0	1728.0
30	1677.0	758.0	1675.0
Mean	1701.8	776.1	1706.4
Standard Deviation	39.1	27.9	47.9

Uncorrected vs. Corrected Pixel Data

Target Size : 747.0 [mm^2]

No.	Pixel Uncorrected	Pixel Corrected
1	705	674.4
2	653	679.1
3	724	684.0
4	720	677.0
5	661	672.7
6	692	673.3
7	658	679.4
8	657	677.4
9	665	681.7
10	714	675.6
11	658	677.5
12	710	680.3
13	675	671.9
14	721	681.1
15	645	681.4
16	720	680.2
17	639	677.9
18	715	678.7
19	713	674.6
20	640	680.9
21	719	678.2
22	660	680.5
23	660	678.5
24	724	681.8
25	643	675.4
26	718	678.3
27	672	675.9
28	720	679.1
29	718	678.3
30	655	681.2
31	664	676.7
32	665	673.8
33	719	678.2
34	647	675.8
35	719	681.4
36	719	680.3

37	662	677.6
38	717	676.3
39	663	676.7
40	645	675.6
41	642	679.2
42	680	671.8
43	646	680.5
44	723	684.1
45	718	676.2
46	643	678.3
47	647	682.5
48	717	676.3
49	651	680.9
50	717	677.3

Pixel Measurements for 4 Different Orientation of Metal Targets

(separate calibration for each side of the belt)

LEFT SIDE of the belt; Area of metal target : 1717 mm²

Target Orientation

No.	Vertical [pixel]	Horizontal [pixel]	- 45° [pixel]	+ 45° [pixel]
1	1562.8	1559.3	1563.0	1544.1
2	1570.5	1560.1	1554.0	1546.2
3	1562.9	1567.7	1568.5	1556.4
4	1564.3	1558.1	1560.3	1560.8
5	1561.8	1564.5	1562.9	1564.0
6	1568.0	1550.1	1567.7	1563.4
7	1558.2	1548.1	1564.0	1557.1
8	1553.6	1561.9	1546.8	1547.5
9	1545.9	1553.9	1562.8	1543.9
10	1558.5	1565.9	1553.0	1565.9

RIGHT SIDE of the belt; Area of metal target : 1703 mm²

Target Orientation

No.	Vertical [pixel]	Horizontal [pixel]	- 45° [pixel]	+ 45° [pixel]
1	1552.3	1550.6	1547.9	1537.9
2	1563.0	1549.4	1542.8	1548.3
3	1549.7	1554.9	1557.2	1543.6
4	1553.7	1546.1	1546.5	1560.1
5	1549.5	1546.7	1546.6	1556.9
6	1562.7	1539.5	1560.2	1560.4
7	1548.2	1540.8	1560.7	1549.4
8	1548.8	1549.6	1537.9	1548.2
9	1535.3	1540.6	1550.6	1543.0
10	1551.7	1555.9	1541.1	1560.7

Data for Comparison of Measurements on the Left and Right Side of the Belt

No.	Metal Target Size [mm^2]					
	3123.0	2806.0	1717.0	1703.0	1564.0	471.0
LEFT SIDE						
1	2887.2	2583.8	1556.9	1575.1	1434.2	425.3
2	2885.5	2589.6	1569.9	1572.8	1427.5	437.6
3	2890.6	2580.3	1562.5	1558.0	1418.7	415.1
4	2888.6	2595.6	1570.1	1568.8	1431.5	434.3
5	2880.5	2588.9	1571.6	1563.0	1434.1	431.5
6	2875.1	2586.5	1571.7	1566.5	1433.8	416.8
7	2898.1	2563.7	1569.6	1564.7	1446.2	432.9
8	2869.9	2593.1	1569.6	1562.0	1425.2	417.1
9	2883.5	2565.4	1563.1	1566.1	1433.9	434.7
10	2877.7	2567.0	1566.3	1558.4	1425.5	439.5
RIGHT SIDE						
1	2820.6	2563.8	1570.6	1532.1	1424.3	417.1
2	2771.9	2557.7	1553.6	1531.0	1413.9	418.5
3	2833.5	2540.5	1522.0	1524.3	1416.2	415.4
4	2815.4	2569.8	1534.0	1537.3	1426.5	414.2
5	2796.6	2573.1	1526.3	1548.2	1411.9	416.3
6	2839.0	2566.6	1545.6	1554.0	1419.0	423.8
7	2788.2	2567.4	1525.5	1530.4	1432.4	414.6
8	2836.5	2559.3	1528.9	1533.4	1420.2	444.7
9	2779.6	2575.3	1523.2	1535.7	1424.1	416.8
10	2792.2	2562.4	1516.0	1530.9	1427.1	442.8

Data for Comparison of Measurements on one Side of the Belt

Area of Metal Target : 1717 mm^2

Position 1 : 11 cm from the center axis of the belt o the LEFT
Position 2 : 7 cm from the center axis of the belt to the LEFT

	Target Position	
No.	Position 1	Position 2
1	1575.7	1575.0
2	1563.1	1567.9
3	1561.5	1576.7
4	1573.2	1561.5
5	1556.1	1566.7
6	1576.5	1579.4
7	1575.8	1557.4
8	1564.5	1564.8
9	1578.6	1563.0
10	1577.2	1575.1

Area of Metal Target : 1703 mm^2
Position 3 : 11 cm from the center axis of the belt to the RIGHT
Position 4 : 7 cm from the center axis of the belt to the RIGHT

	Target Position	
No.	Position 3	Position 4
1	1563.0	1564.1
2	1561.7	1547.7
3	1542.1	1560.6
4	1545.7	1561.0
5	1559.2	1548.8
6	1544.7	1550.8
7	1547.0	1556.6
8	1563.1	1542.4
9	1551.8	1563.6
10	1550.1	1553.1

Pixel Measurements in Three Different Speeds

LEFT SIDE of belt; Target Size : 1027 mm^2;

No.	Stationary	Speed [cm/s] 30 cm/s	40 cm/s
1	903.4	904.9	913.0
2	903.0	908.8	910.6
3	899.0	917.1	908.0
4	913.9	908.1	901.6
5	906.2	902.3	899.2
6	908.8	901.8	908.1
7	901.1	902.7	902.1
8	910.0	903.0	915.6
9	911.4	905.4	901.7
10	905.3	902.3	903.9
11	905.9	907.0	900.0
12	906.7	896.5	903.8
13	902.7	900.6	907.3
14	900.1	899.8	905.0
15	901.0	905.1	904.0

RIGHT SIDE of belt; Target Size : 2207 mm^2;

No.	Stationary	Speed [cm/s] 30 cm/s	40 cm/s
1	1975.1	1978.3	1995.1
2	1990.6	1992.6	1991.6
3	1976.9	1989.1	1985.5
4	1995.7	1993.8	1991.6
5	1990.2	1992.4	1996.7
6	1987.8	1989.7	1998.4
7	1988.0	1991.8	1975.4
8	1982.1	1972.0	1983.5
9	1987.4	1977.6	1984.1
10	1988.5	1991.5	1977.9
11	1989.8	1989.5	1996.2
12	1985.9	1982.8	1975.0
13	1988.1	1987.1	1990.0
14	1978.1	1970.8	1983.3
15	1971.4	1987.1	1987.0

Area Calibration Data for 26 Metal Targets

(separate calibration for each side of the belt)

Target No.	LEFT SIDE Area [mm^2]	LEFT SIDE Pixel [pixel]	RIGHT SIDE Area [mm^2]	RIGHT SIDE Pixel [pixel]
1	891.0	824.8	1175.0	1066.9
2	1175.0	1069.5	891.0	820.2
3	2233.0	2055.3	1564.0	1419.1
4	1564.0	1437.5	2233.0	2025.1
5	3123.0	2882.4	1856.0	1696.4
6	1856.0	1728.7	3123.0	2831.3
7	1871.0	1717.1	1386.0	1273.6
8	1386.0	1265.6	1871.0	1674.8
9	1050.0	968.2	1658.0	1522.3
10	1658.0	1545.6	1050.0	974.6
11	2806.0	2600.4	471.0	417.7
12	471.0	445.4	2806.0	2536.9
13	747.0	690.1	2475.0	2253.9
14	2475.0	2305.2	747.0	699.5
15	1889.0	1709.0	1889.0	1731.0
16	2410.0	2174.0	1200.0	1128.0
17	1200.0	1090.0	2410.0	2124.0
18	1560.0	1478.0	1535.0	1483.0
19	1535.0	1432.0	1560.0	1455.0
20	1703.0	1542.0	1027.0	915.0
21	1027.0	981.0	1703.0	1641.0
22	1717.0	1548.0	2580.0	2307.0
23	2580.0	2319.0	1717.0	1520.0
24	2207.0	1993.0	550.0	488.0
25	550.0	503.0	2207.0	2013.0
26	2975.0	2693.0	1918.0	1745.0

Uncorrected Pixel Data for Oyster Meats

Target Size		
	Left :	11.954 [g] Oyster Meat
	Right :	19.531 [g] Oyster Meat
	Center :	731.0 [mm^2] Metal Target

No.	Target Position		
	Left Oyster	Center	Right Oyster
1	1301.0	691.0	2094.0
2	1421.0	688.0	2194.0
3	1292.0	670.0	2388.0
4	1418.0	685.0	2181.0
5	1309.0	687.0	2237.0
6	1265.0	625.0	1870.0
7	1282.0	657.0	2438.0
8	1394.0	688.0	2226.0
9	1302.0	686.0	2085.0
10	1267.0	629.0	2090.0
11	1279.0	672.0	2385.0
12	1512.0	690.0	2209.0
13	1269.0	626.0	1945.0
14	1418.0	684.0	2289.0
15	1283.0	635.0	2262.0
16	1269.0	626.0	1668.0
17	1261.0	628.0	2356.0
18	1403.0	687.0	2220.0
19	1262.0	628.0	2211.0
20	1282.0	643.0	2586.0
21	1293.0	677.0	2143.0
22	1283.0	666.0	2375.0
23	1308.0	683.0	2212.0
24	1261.0	623.0	1892.0
25	1270.0	631.0	2096.0
26	1280.0	642.0	2097.0
27	1289.0	643.0	2436.0
28	1266.0	630.0	2247.0
29	1266.0	627.0	2006.0
30	1277.0	666.0	2375.0
Mean	1309.4	657.10	2193.77
Standard Deviation	64.10	26.21	193.26

Weight/Area Calibration Data for 54 Oyster Meats

(separate calibration for each side of the belt)

Measurement No.	LEFT SIDE Area [pixel]	LEFT SIDE Weight [g]	RIGHT SIDE Area [pixel]	RIGHT SIDE Weight [g]
1	1443.0	11.891	1593.0	16.807
2	1084.0	9.258	1327.0	13.001
3	1456.0	12.238	1382.0	12.835
4	1035.0	9.158	1574.0	15.655
5	867.0	6.631	937.0	9.700
6	1347.0	11.146	1287.0	13.409
7	1324.0	10.669	1139.0	9.035
8	1110.0	7.903	935.0	8.323
9	918.0	6.642	831.0	7.704
10	1687.0	14.942	1802.0	18.582
11	1461.0	12.345	1831.0	16.579
12	1475.0	14.227	1051.0	8.958
13	1108.0	10.588	1333.0	12.926
14	1336.0	11.498	1208.0	11.513
15	951.0	6.711	1302.0	10.968
16	1289.0	12.030	1036.0	7.176
17	2084.0	17.131	983.0	9.939
18	991.0	8.361	1665.0	18.406
19	1529.0	11.359	1457.0	15.031
20	1277.0	11.699	1199.0	9.427
21	893.0	5.266	1418.0	13.424
22	1731.0	14.782	1516.0	12.035
23	861.0	8.515	741.0	5.333
24	904.0	6.107	728.0	5.197
25	1689.0	16.163	1347.0	13.881
26	862.0	6.152	1197.0	12.572
27	1268.0	10.930	831.0	7.405

The vita has been removed from the scanned document