**MVA'90** IAPR Workshop on Machine Vision Applications Nov. 28-30,1990, Tokyo

# Defect Detection, Classification and Quantification in Optical Fiber Connectors

**Behzad Shahraray**
**Arthur T. Schmidt**
*Machine Perception Research Department*
AT&T Bell Laboratories
Holmdel, NJ 07733

**J. Mark Palmquist**
*Lightguide Apparatus Process Development Department*
AT&T Network Systems
2000 NE Expressway
Norcross, GA 30071

## ABSTRACT

**A machine vision system for the detection, classification and quantification of defects in polished surfaces of optical fiber connectors is described. These polished surfaces can contain several classes of defects such as chips, pits, scratches and cracks. Such defects can potentially lead to malfunction and may go undetected by a functional test measuring the attenuation of the light signal passing through the connector. The detection of such defects involves visual inspection of the connector end surfaces using microscope images. An automatic method for the determination of three focus locations for obtaining images which collectively contain all the features of interest is developed. Information from these reflected-light images is combined into a single image on which further processing is done. The discrepancies between these images are used to discriminate between cracks and scratches. Morphological processing is used to segment the fiber from its ceramic surround. Hough transform is then employed to obtain an estimate of the coordinates of the center of the fiber as well as its radius. This provides a robust method for isolating the fiber even when large portions of it are missing. Chips and pits in the fiber are detected and quantified by a combination of thresholding and morphological processing of the isolated fiber. A specialized edge detector is used to detect edge segments resulting from scratches and cracks present on the fiber end. The detected segments are then matched and combined to form the scratches and cracks. Matching the line segments is complicated by the presence of parallel scratches and cracks in proximity of each other. A line matching algorithm capable of dealing with this situation is developed. The system provides quantitative measures which are used to assess and improve the effectiveness of the polishing process and to establish standards for the quality of the connectors. Once a criterion for the acceptability of the fiber connector has been established, the system can be used to perform the final inspection.**

## I. INTRODUCTION

The advent of optical fiber communications has brought special significance to the cost effective production of high quality fiber optic components. Manufacturers of lightguide interconnect cables strive to provide low insertion loss connectors with predictable performance.

Assembly of a single fiber lightguide cable into a connector typically requires the use of epoxy to adhere the bare fiber to the bore of the connector. Once the epoxy has been cured, the end face of the connector is ground and polished to remove excess adhesive and provide a smooth, flat or slightly convex surface for light to enter or exit the fiber at the interface between connectors. A single-mode fiber has an approximate diameter of 125 micrometers. However, the core of the fiber which is responsible for the light propagation is only eight micrometers in diameter. The presence of dirt, cracks, chips, scratches or pits in the core region can cause increased insertion loss or lead to premature failure due to propagation of cracks under variable environmental conditions or mechanical stress.

As part of the manufacturing process, the assembled connectors are subject to a functional test which measures the insertion loss, thereby detecting anomalies in the core region. Such a functional test, however, may fail to identify cosmetic defects indicative of process problems, or detect potential sources of problem such as cracks which when propagated can lead to failure, Consequently, all connectors are visually inspected under microscopes with magnification ranging from 50X to 200X.

Visual inspection is a subjective task requiring intense concentration and attention to detail. Results can vary from one operator to another and little or no quantitative information can be reliably obtained to support statistical process control of the polishing process. The lack of consistent quantitative measures for the defects has been a major obstacle for the development of standards for the quality of the connectors. As a result, no industry standards for inspection of polished connector end faces exist.

The need to improve quality and reduce costs has created a need to develop an automated inspection procedure capable of locating and classifying possible defects as well as providing quantitative information on the overall quality of the surface finish. This paper presents a machine vision system developed to meet these needs. Proper use of this quantitative information enables process engineers to quickly pinpoint causes of problems, such as defective polishing media, incorrect machine settings or improper operator technique. Automated inspection relieves production specialists of having to deal with the problems associated with manual inspection and provides them the necessary information to improve the process.

The *Area of Interest* in the reflected light image of the connector end is the glass fiber which has a circular

cross section which may be severely affected by the existence of chips in the boundary. There is a thin band of epoxy between the fiber and the ceramic surrounding which is an artifact created by the "protruding fiber" polishing method. The possible defects can be put into several different categories based on their properties as follows. *Chips* are voids in the surface which extend to the boundary of the fiber. When illuminated by incident white light from a microscope, chips appear darker than the surrounding smooth fiber surface. *Pits* are voids which occur inside the fiber and are completely enclosed by the fiber. Sometimes the polishing process causes a shard of material to roll across the surface and create an array of pits along a line. These are called *digs*. The process of polishing uses abrasives to remove material. The path left by the abrasive is usually a straight line referred to as a *scratch*. Successive polishing steps produce smaller and smaller scratches until the desired degree of surface finish is obtained. Sometimes defective or contaminated polishing media will create a new scratch in a smooth surface. Scratches are usually very straight, and have a higher average intensity than the surrounding area. Scratches may be only a few nanometers deep but still constitute a cosmetic surface defect. *Cracks* are a separation of the material, and have a unique property that their intensity profile changes considerably when viewed slightly above or below the focal plane. Cracks in fibers are usually curved and may branch. Cracks have essentially zero width and may not be visible when viewed in-focus under high magnification. The structure or presence of a crack often becomes apparent if the microscope is slightly out of focus. Cracks pose a real problem since in time they can propagate and cause the fiber to fail.

The remainder of the paper is organized as follows. A method for the acquisitions and fusion of images at different focus positions is discussed in section II. Section III presents a method for identifying the area of interest. Algorithms for the detection of chips, pits, scratches and cracks are presented in sections IV and V. Some implementation issues are discussed in section VI.

## II. IMAGE ACQUISITION

The quality of the images on which analysis is performed has a major impact on the final results of the process. The two most significant factors governing the image quality are lighting and focusing. Because of the small size of the fibers, their visual inspection requires magnification. Magnification is obtained through the use of a microscope.

The property of the optical fibers in focusing the major portion of the light passing through the fiber towards the core of the fiber rules out the use of back lighting. Reflected-light images perform much better in bringing out the surface properties of the polished surface of the fiber. The high magnification involved calls for an intense and highly focused illumination source. This is provided by a focused beam passing through the same objective used in image acquisition.

The light level can be adjusted to allow full utilization of the dynamic range of the sensor, while avoiding the possibility of sensor saturation. Some variability in the lighting conditions can be compensated during processing by adjusting the parameters based on the grey-level histogram. The variability in focusing, on the other hand, can give rise to significantly different results and is difficult to deal with once the image has been acquired. In fact, when performing the inspection manually, changes in the focusing from one trial to another (with the same operator or two different operators) is a major obstacle in obtaining consistent results. Similar problems exist with semi-automated inspection using manual focusing. These problems stem from the extremely limited depth of focus of microscope images. The limited depth of focus affects the results in two ways. First, the lack of repeatability when the focus position is determined manually by an operator leads to inconsistent results when the same fiber is focused, digitized and processed several times. This is because a slight change in focus can alter the features considerably. Second, different features of interest do exist at different planes of focus. This fact coupled with the extremely limited depth of focus makes it impossible to obtain an image in which all of the features of interest are in sharp focus.

The first problem can be rectified by using an automatic focusing scheme. Such a step goes a long way towards reducing the variability. However, the second problem persists. One way of dealing with the second problem is to obtain images at several different focus settings and to fuse the information into a single image. We will discuss both problems and present solutions to them.

### II.a. Automatic Focusing

Performing automatic focusing requires computer control of the microscope focus. This is achieved by equipping the microscope focus mechanism with a stepper motor which can be positioned under program control. Given sufficient resolution in controlling the focus, we require a metric (or "focus function") to measure the sharpness of focus.

Several criteria for computing a focus function have been proposed in the literature (see [1]). These include measures based on grey-level variance, intensity gradient, high pass filtering, Fourier analysis and histogram entropy. Experiments with several of these indicate that focus functions based on intensity gradient lead to better results for our application. This is consistent with the results reported in [1] in which several methods are considered. One major difference between our results obtained using microscope images of optical fibers, and the results of [1] is that [1] reports a unimodal function, whereas the focus function obtained here has two closely spaced peaks.

Since the image contains features which are at different focus planes, computing the function on the entire image will affect the focus function adversely. It is advantageous to base the computations on a smaller portion of the image. Our experiments indicate that using a small region of the image ($100 \times 100$ pixels) located on the ceramic surround generates good results.

The gradient function at each pixel inside the window is computed by convolving the image with two $7 \times 7$ tensor product smoothing spline filters. These filters generate estimates of the first derivative of the intensity function in $x$ and $y$. The two derivatives are thresholded to reduce the effects of noise, and used to compute the squared gradient function

$$|\nabla I(x,y)|^2 = I_x^2 + I_y^2. \qquad (2.1)$$

As mentioned before, the focus function obtained this way is bimodal. In fact when looking at the ceramic

surround under the microscope the scratches come into sharp focus at two distinct positions. The position at which scratches on the surface of the fiber come into sharp focus does not coincide with either of these peaks. However, this position can be determined based on the location of the two peaks. In most cases, images obtained at this focus position do not provide sufficient information about the cracks.

### II.b. Multiple Image Fusion

The limited depth of focus of the microscope poses problems in obtaining images with sufficient information about all features of interest. As mentioned before, often the image which best represents the scratches is not equally good for finding cracks and defining the borders of chipped areas. Furthermore, in most cases it is difficult to distinguish between cracks and scratches based on a single image.

Cracks are much deeper than scratches and affect the optical properties of glass in a different way. Moreover, since illumination is accomplished by a beam passing through the microscope objective, changing the focus position also tends to modify the illumination. Consequently, cracks exhibit a much larger degree of variation in grey-level under changes in focusing, especially when changes are made around the position of best focus for scratches. This property is utilized to detect them and distinguish them from the scratches.

Once the optimum focus position for scratches is found, the focus function is used to determine two auxiliary focus positions on opposite sides of the optimum position. Three images obtained at the optimum (primary image) and auxiliary (auxiliary images) focus positions are then used to generate a single image as described below.

An image is formed by computing the absolute value of the difference between the two images obtained at auxiliary focus positions. The difference image is thresholded to eliminate small differences resulting from minor variations between the two images. The binary image generated is further refined through morphological closing followed by opening both with circular structuring elements [2],[3]. The size of the disks are chosen such that components within a predetermined distance are merged and components under a certain size are eliminated. The processed difference image is then smoothed by a low-pass filter with gain $G$ to generate a grey-scale image. $G$ is chosen such that it brings the intensity of the generated image to typical levels of scratches. The low-pass property of the filter generates a decreasing intensity profile which gradually approaches the zero background.

The final image is formed by setting each pixel to the larger of the values of the corresponding pixels in the primary image and the processed difference image. As a result most of the pixels in the primary image are retained, while the regions where cracks occur are replaced by the grey-scale difference image. The binary difference image is retained for discrimination between cracks and scratches. This will be further discussed in section V.

## III. FIBER ISOLATION

We are only interested in the features of the fiber, not those of the its surrounds. Therefore, before proceeding to find the features we should isolate the fiber in the image. This requires the estimation of the center and the radius of the fiber.

First, histogram-based thresholding is used to partially segment the fiber from the background. The segmentation obtained this way is not complete. This is due to the presence of a ring of epoxy around the fiber which is part of the manufacturing process and has grey-levels similar to those of the fiber. Moreover, some of the features on the ceramic have grey-levels in the same range as the fiber. The thresholded image of the fiber, however, is far more solid on the fiber than on the epoxy and the ceramic. This feature is used to improve the segmentation by a morphological opening of the binary image with a disk. The opening operation eliminates a large portion of the features in the background and the epoxy, and breaks up the remaining unwanted features into small, disjoint regions. The unwanted regions are eliminated by performing connected component labeling, computing the size of each connected component and eliminating those which are small.

The image obtained from the segmentation process is often not a perfect disk. There are two major reasons for this. First, the segmentation process can either include part of the background or remove a defective portion of the fiber with significantly different intensity. Second, chips can cause large portions of the fiber to break off. In these cases, the result of segmentation is a partial disk with imperfect and partial boundaries. The existence of partial data makes the application of template matching unattractive. Instead we use a method based on the Hough transform discussed below.

Given a circle, it is known that the perpendicular bisector of every chord passes through the center of the circle. Hence, in order to find the center of the circle it is sufficient to find the intersection of the perpendicular bisectors of two arbitrary (non-parallel) chords. Once the center has been found the radius can be obtained by computing the distance of the center from any point on the circle. Given a disk with noisy and imperfect boundaries, this property can be used to estimate its center by considering a large number of pairs of points on the boundary and computing the perpendicular bisectors of the chords which go through the pair. A two-dimensional accumulator can then be used to keep track of the points on the bisectors. Given sufficient number of good points on the boundary of the circle, an estimate of the center can be obtained by finding the accumulator cell with the maximum number of votes. A similar voting scheme can then be used to estimate the radius by incrementing a one-dimensional discrete accumulator based on the distance of each boundary point from the estimated center. The value associated with the accumulator cell with the maximum number of votes is the estimated radius of the circle.

The above algorithm was applied and generated good estimates of both the coordinates of the center and the radius of the fiber. A two-dimensional accumulator discretized to the resolution of the image, and limited to a rectangle about the center of the image was used. A point inside the fiber (closer to the boundary than to the center of the fiber) was used to generate a number of uniformly spaced rays. Each ray was traced from its two endpoints (points of intersection with the image boundary) until the first nonzero pixel was found. The two points found this way were taken as points on the boundary of the segmented image and were used to compute

the perpendicular bisector of the chord.

The method can be made more robust by using more than one point as the source of rays used to obtain the chords. This will help reduce the possibility of generating too many bad chords when the source of rays happens to be close to part of the boundary with a major defect. Considerable reduction in computational complexity of the method can be made by restricting the rays to horizontal and vertical directions. This will cause the rays to coincide with the rows and columns of the image, and leads to major simplifications in computing the bisectors. Moreover, since the horizontal and vertical chords can only vote for the $x$ and $y$ coordinates of the center respectively, we can use two one-dimensional accumulators instead of a two-dimensional one. It should be pointed out that the reduction in the computational complexity is at the expense of robustness. When using rays of arbitrary direction generated from several points the method is capable of dealing with distortions which alter a large portion of the boundary.

The algorithm does not use information about the radius of the fiber. However, once the radius has been estimated, it is checked against the known radius. A large discrepancy is indicative of a major problem with the fiber and is used to terminate the process.

The estimates of center and the radius of the fiber are used to isolate the fiber by clearing all the image pixels outside the fiber. The size of the image is reduced to contain only the fiber and a border sufficient to allow performing neighborhood operations on the image.

## IV. DETECTION OF CHIPS AND PITS

Among the possible defects on the polished surface of the fiber, chips and pits are the easiest to detect. These defects shows up as regions with intensities which are markedly different from the rest of the fiber. Therefore, they can be separated by thresholding.

The histogram for the isolated fiber is used to determine the range of grey level for the fiber. Thresholding is used to separate the regions outside this range. The image obtained this way includes additional unwanted features due to noise, scratches and cracks. Moreover, an individual chip or pit may be broken into several pieces because of intensity changes inside the chip or pit region. Morphological closing by a disk is used to merge close segments and fill small holes inside the detected regions. The resulting image is then opened by a disk. The size of the disk is adjusted such that it is sufficiently large not to fit inside the segments resulting from cracks, scratches and small regions which are due to noise. The opening process eliminates the scratches and cracks because of their limited thickness. It also performs size filtering by eliminating chips which are under a certain size threshold determined by the size of the disk structuring element used.

Connected component labeling is used to mark individual chips and pits. Further size filtering is performed by computing the size of each component and deleting the ones below a predetermined size threshold. The image which is input to the connected component labeling stage is also retained for use by the scratch/crack detection stage which will be discussed in the next section. Various statistics about chips and pits such as size, center of mass, distance of center of mass form the core, minimum distance from the core and number of pixels in the restricted area are then computed.

## V. DETECTION OF SCRATCHES AND CRACKS

Detection of fine scratches and cracks with insufficient contrast is a difficult task. Performing this task involves the application of an edge detector. Since we are interested in detecting a particular class of edges, it is possible to use a specialized edge detector.

Because we are also interested in quantification rather than detection alone, some steps have to be taken in order to insure the consistency of the measurements under variations in the imaging conditions. First, the histogram for the isolated image is used to determine the range of interest. The image intensity values in the given range are then stretched to fill the entire range. Image intensity values below and above the range are set to the minimum and maximum values respectively. This allows us to get consistent measurements without adjusting the parameters in subsequent stages of processing.

### V.a. Edge Detection

A rather specialized detector is used to find fine scratches and cracks. Given a scratch of finite width, we want to find an edge located at the center (along the width) of the scratch. The maxima of the second order derivative of intensity is used to locate these edges. This is achieved by applying local operators to estimate the derivatives along four directions (horizontal, vertical and two diagonals). Prior to the application of directional derivative operators the image is smoothed. The smoothing operation serves two purposes. First, it reduces the effects of high frequency noise. Second, if there are scratches with a flat intensity profile (caused by sensor saturation, quantization, wide scratches, etc.), smoothing will convert them such that their intensity profile is bell-shaped. This will enable the derivative operators to generate a peak corresponding to the center of the scratch. A $7\times7$ smoothing spline operator is used to perform the smoothing. The second derivatives of the image are computed along the horizontal, vertical and diagonal directions using $5\times5$ directional derivative operators. The magnitude of the four directional derivatives are then compared at each point to generate the edge strength and direction images as described below.

The edge strength image is computed by setting each pixel of the strength image to the value of the corresponding pixel in one of the four directional derivative images which has the maximum absolute value. Since scratches are brighter than the fiber, the strength image computed this way will have a negative peak at the center and two positive peaks on opposite sides of the scratch. Cracks, on the other hand, are usually darker than the background and sometimes have both a dark and a bright side. However, when using the combined image, cracks behave similar to scratches. The distinction between cracks and scratches, can still be made based on the processed difference image. For detection purposes it is sufficient to retain the pixels with negative maximum (in absolute value) second derivative. The threshold is set below zero to eliminate edge pixels arising from extremely weak features or noise and leave pixels with negative derivatives of sufficiently large magnitude. Figure 3 shows the edge strength image after thresholding.

Information about the direction of edge pixels is computed and stored in an edge direction image. Given that the derivative magnitude is maximum in the

direction perpendicular to the edge, a direction image quantized to four directions can be computed by simply recording which of the four directional derivative images is maximum at a given pixel. Such a coarse quantization of the edge direction, however, proved to be inadequate for edge tracing and resolution of intersecting edge segments. In order to obtain edge direction information with more levels of quantization the following method is employed.

Three out of four directional derivatives are used to compute the coefficients of an interpolating quadratic polynomial by taking the maximum derivative as the center point (i.e., zero) and the values of the derivatives on both sides of it as the other two points (set to -45 and 45). For example, if the second derivative is maximum at the 45 degree direction, it is taken as the center point (zero), while the the derivatives in the horizontal and vertical directions are taken as -45 and 45 respectively. Once the interpolating polynomial is computed, the location of the peak (maximum or minimum) is the offset of the direction of maximum second derivative from the center point. The fixed distance of the points used in the curve fitting allows for the efficient computation of the peak with only a few arithmetic operations. The direction information thus obtained is quantized to forty levels and stored in the direction image.

Next, the strength and direction images are used to perform nonmaximum suppression and thinning of the edge segments [4]. These operations are performed only on the strength image although the direction image is used in the process.

### V.b. Edge Tracing and Decomposition

The edge images obtained from edge detection contain a large number of small segments resulting from noise. Moreover, there are spurious segments resulting from the fiber boundaries and boundaries of chips and pits. Before attempting to label the edges, the information about the center and radius of the fiber as well as information about the detected chips and pits is used to eliminate the spurious segments by masking the edge strength image with dilated images of the detected chips and pits and the fiber edge image. Next, a fixed threshold is applied to the strength image to eliminate weak edge segments. A thresholding with hysteresis such as the one reported in [4] would improve the results. However, simple thresholding is used to reduce the computational requirements. Segments due to noise above the threshold level are dealt with after decomposition and labeling.

Because of the presence of overlap between scratches, cracks, pits and chips, edges can be broken into several segments. Furthermore, segments from different scratches and/or cracks can give rise to composite connected edge segments. Before correspondence can be established between separate segments belonging to the same feature, it is necessary to identify connected segments and decompose the composite segments into the separate components. The presence of the composite segments rules out the use of a connected components labeling algorithm since labeling based merely on connectedness will fail to decompose these segments. However, by considering the fact that in most cases the transition from one feature segment to another is not smooth, it is possible to utilize the edge direction information in the labeling process to separate the individual components.

The labeling and decomposition task is performed by an augmented connected labeling algorithm which groups the pixels based on connectedness but also takes into account the edge direction information. By allowing limited changes in edge direction from one edge element to another, the algorithm is capable of following curved edges while being able to break these edges at sharp corners. This also enables the algorithm to deal with fork points by following the path with the least change in the direction as long as the change is within the predetermined limit. The predetermined threshold on the direction governs the tradeoff between the maximum curvature of a traceable segment and the effectiveness of the algorithm in decomposing composite segments. In general, problems due to an edge segment being labeled with multiple labels are less catastrophic than the ones arising from composite segments which receive the same label. Once the edge segments have been decomposed and labeled, small segments due to noise are eliminated based on their size. The resulting image is shown in figure 4.

### V.c. Classification

Because of the serious nature of the problems caused by cracks, it is important not only to detect them, but to be able to distinguish them from scratches. In order to avoid matching segments of features belonging to different classes (i.e., scratches and cracks), classification of features must be performed prior to line matching.

The processed binary difference image is used to perform the classification. Each segment resulting from the edge tracing and decomposition is classified as a crack if the majority of the corresponding pixels (i.e., pixels at the same location) in the difference image are non-zero. Otherwise, the segment is classified as a scratch. The line matching stage operates on straight scratches and straight cracks separately.

### V.d. Line Matching

The line features obtained as a result of applying the edge detector to the image followed by component labeling are often broken into several segments. It is necessary to group segments belonging to the same feature together before quantitative information about the feature can be computed. In this section we present an algorithm for matching straight line segments belonging to the same line.

Scratches make up the major portion of the edge features present on the fiber. Because of the nature of the polishing process and the small size of the fiber, the majority of scratches are straight lines. Therefore, the line matching algorithm is applicable to a large percentage of the scratches. Cracks, on the other hand, may not be straight. Consequently, the line matching method may not be applicable to some of the cracks. The ability of the component labeling method to follow curved lines reduces the number of segments a curved crack is broken into. At this point, no attempt is made to match curved line segments. Such features, if present, are quantified individually.

First, the components are converted to a higher level representation which is suitable for matching. A straight line identified by its orthogonal distance from the origin (denoted by $d$), and the angle it makes with the $x$-axis ( denoted by $\theta$) is fitted to each component

using least squares approximation, minimizing the orthogonal distance from the line. The end-points of the fitted line are computed by projecting the end-points of the data into the fitted line. The mean of squared differences between the data and the approximating line is used to separate straight segments from curved segments by applying a fixed threshold. Curved segments are excluded in the matching process.

Theoretically, the $d-\theta$ representation would be sufficient for establishing the correspondence between the isolated straight line segments. However, several practical issues reduce the validity of matches based merely on the computed $d-\theta$ values. In the presence of noise, estimates of $d$ and $\theta$ based on a small number of points may differ from the exact parameters of the line segment. This should be taken into account when comparing the parameters of lines, by allowing lines with close parameters to match. However, doing so will cause distinct parallel lines with close $d$ values to match. This is undesirable because of the existence of close but distinct parallel scratches. Another problem which arises is due to the fact that the $d-\theta$ representation deals with infinite lines and does not carry any information about the length and the position of the line it is representing. Therefore, two short and very distant (along the direction of the line) line segments with no segments between them can be matched. A line matching method using a distance measure which is sensitive to closely spaced parallel lines was developed to deal with the above problems.

Let $\phi_m$ and $\phi_n$ denote the angles that the line connecting the centers of lines $m$ and $n$ make with line $m$ and line $n$ respectively. Then the distance $D_{m,n}$ between line $m$ and line $n$ is defined as

$$D_{m,n}=\mathrm{MAX}(\phi_m,\phi_n). \tag{3.1}$$

The distance measure is depicted in figure 1. The match coefficient is defined using the distance measure as,

$$M_D(m,n)=\begin{cases} 0 & \text{if } \delta_d(d_m,d_n)>d_{th} \quad (I) \\ & \text{or } \delta_\theta(\theta_m,\theta_n)>\theta_{th} \quad (II) \\ & \text{or } D_{m,n}>D_{th} \quad (III) \\ 1-\dfrac{D_{m,n}}{D_{th}} & \text{otherwise} \end{cases} \tag{3.2}$$

where, $\delta_d(d_m,d_n)$ and $\delta_\theta(\theta_m,\theta_n)$ are the difference between the $d$ and $\theta$ parameters for the two line segments, and $d_{th}$ and $\theta_{th}$ and $D_{th}$ are predetermined thresholds on $\delta_d(d_m,d_n)$, $\delta_\theta(\theta_m,\theta_n)$ and $D_{m,n}$ respectively. Conditions I, II and III are checked sequentially to reduce the need for further computations. The measure is most sensitive to non-colinear line segments whose centers are close to each other. In these cases, the line connecting the centers of the two line segments makes large angles with the line segments. As the centers of gravity of the line segments move away from each other, the distance measure decreases which leads to a larger match coefficient. In order to control this property effectively, and prevent the matching of lines which are too far from each other, a penalty is imposed on the distance measure using a correction factor $CF(m,n)$ based on the ratio of the length of the line connecting the centers (denoted by $l$) to the sum of the lengths of the line segments.

$$M(m,n)=M_D(m,n)\ CF(m,n). \tag{3.3}$$

The correction factor is defined as,

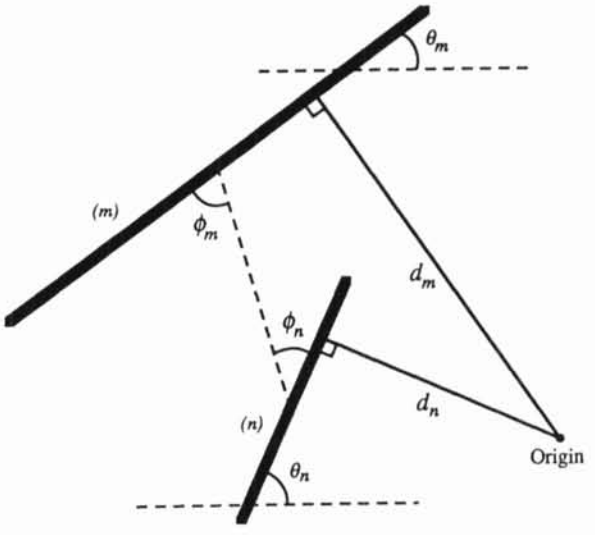


**Figure 1.** ( $d$, $\theta$ ) line representation and the $\phi$ angles used to compute the distance measure.

$$CF(m,n)=1-\alpha\left[\frac{L(l)}{L(m)+L(n)}\right], \tag{3.4}$$

where $L(.)$ denotes the length of the line and $\alpha$ is a constant which controls the significance attached to the length ratio.

The line matching method discussed does not depend heavily on the accuracy of the match coefficients in deciding whether the line segments are colinear. Instead, it utilizes the information about the pair-wise match coefficients to iteratively find the best possible matches, and to propagate the information about pair-wise mismatches to other line segments.

Given $N$ line segments, line matching is performed by constructing an $N\times N$ match coefficient matrix $\mathbf{C}$,

$$\mathbf{C}=\{c_{i,j}\ ,\quad i=1,...,N,\quad j=1,...,N\}, \tag{3.5}$$

where the initial values for the match coefficients are given by

$$c_{i,j}=c_{j,i}=\begin{cases} 1 & i=j \\ M(i,j) & i\neq j \end{cases} \tag{3.6}$$

In the first iteration, the coefficient matrix is searched for the largest (smaller than 1) coefficient $c_{p,q}$. If $c_{p,q}$ is larger than a predetermined match threshold $C_{th}$, lines $p$ and $q$ are marked matched by setting

$$c_{p,q}=c_{q,p}=1. \tag{3.7}$$

Subsequently, the pairwise match coefficients of each of the two lines with all the other lines are updated by setting the corresponding coefficients equal to the minimum of the two, i.e.,

$$c_{p,i}=c_{i,p}=c_{q,i}=c_{i,q}= \tag{3.8}$$

$$\begin{cases} 1 & \text{if} c_{p,i}=1 \quad \text{or} \quad c_{q,i}=1 \\ \mathrm{MIN}(c_{p,i},c_{q,i}) & \text{otherwise.} \end{cases}$$

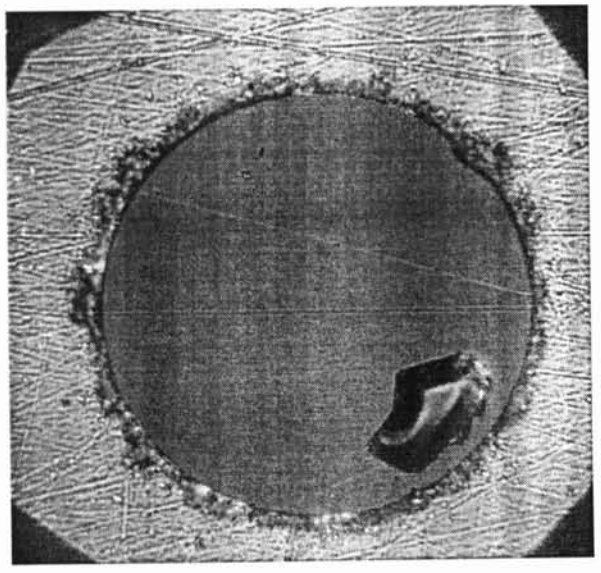

**Figure 2.** Original fiber connector end-face image showing the epoxy ring and the ceramic surrounds. The crack cannot be seen at this focus position.

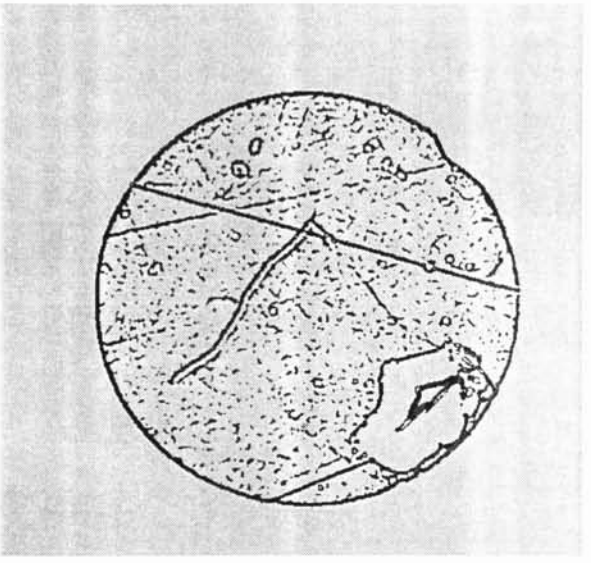

**Figure 3.** Edge strength image resulting from directional derivative computation on the combined image. No non-maximum suppression and thinning performed.

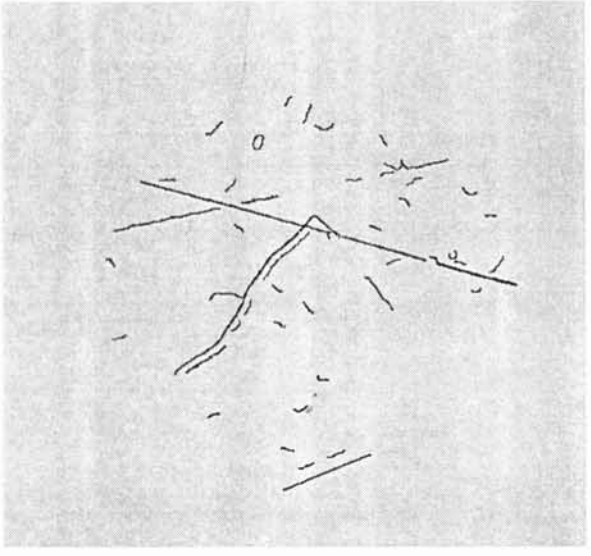

**Figure 4.** Scratch and crack segments after labeling and size filtering.

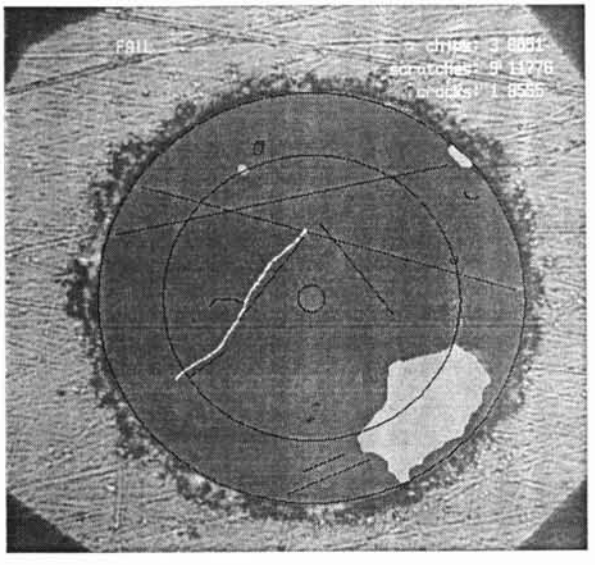

**Figure 5.** Final result of processing superimposed on the fiber image. Cracks are shown in white, scratches are shown in black, chips and digs are shown in grey.

As a result, any mismatch between line $p$ and a third line is propagated to line $q$. Additional iterations are performed by finding the next largest match coefficient (i.e., smaller than or equal to $c_{p,q}$) in the updated matrix and repeating the process. However, the process of updating the match coefficients is more involved than given in (3.8) since a match may be made between a single line and a group of already matched lines, or two groups of lines. Therefore it is necessary to update the coefficients of all of the lines involved, by setting them equal to the smallest of the corresponding coefficients in all of them. The iterative process terminates when the next largest match coefficient is smaller than $C_{th}$. It should be pointed that because of the symmetry of the $M$ matrix, one only needs to construct and update half the matrix. The algorithm description, however, was given based on the whole matrix for the sake of simplicity.

By performing the matches in order and propagating the information about conflicting matches the algorithm is able to group segments of straight lines while avoiding incorrect matched between close, but distinct lines. The correction factor $CF(m,n)$ given by (3.4) serves two purposes. First, it compensates for the fact that the distance measure $D_{m,n}$ tends to decrease as the line segments get more distant. Second, by overcompensating $M_D(m,n)$ for distance (i.e., larger values of $\alpha$ it can impose an order in the matching process which favors matching closer segments first. The more distant lines are eventually matched if they are not in conflict with existing matches.

At the termination of the matching process, each group of matched lines is combined into a single feature. Quantitative information about each feature (such as strength, size, orthogonal distance from the core, etc.) are the computed.

## VI. DISCUSSION

The algorithms described the preceding sections have been implemented on an IRI SP512 vision system. The vision system includes an iconic processor which supports linear and nonlinear neighborhood operations such as convolution and binary morphology.

The hardware limitations in terms of computing power and the type of operations supported, and the constraints imposed on the total processing time played a major role in the algorithm development phase. The processing time for some of the operations such as line matching is data-dependent. Therefore, the total processing time varies from one connector to another. The total processing time for a typical connector is about thirty seconds, most of which is spent on edge detection and line matching. The result of processing for the connector depicted in figure 2 is shown in figure 5. The detected features are superimposed on the original image. The three concentric circles, from large to small, correspond to the fiber, the critical region (defined as a fixed percentage of the fiber area) and the core of the fiber. Black lines represent scratches. The cracks appear in white. Detected chips and digs are shown as grey regions.

An interface is provided which enables the user to access detailed quantitative information about each feature at various stages of processing and use this information to adjust the parameters. The interface also provides a mechanism to define the criteria for acceptability of the fiber connector based on the attributes the features present on the fiber. Once the criteria have been defined, the system can be used to make a binary decision on the quality of the connectors.

## ACKNOWLEDGMENTS

The authors wish to thank Homer Chen and Jakub Segen for many valuable discussions, and Kicha Ganapathy for his support throughout this work.

## REFERENCES

[1] E. Krotkov, "Focusing," *International Journal of Computer Vision*, Vol. 1, No. 3, pp. 223-237, 1987.

[2] J. Serra, *Image Analysis and Mathematical Morphology*. Academic Press, New York, 1982.

[3] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image Analysis Using Mathematical Morphology," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 4, pp. 532-550, July 1987.

[4] J. Canny "Finding Edges and Lines in Images," *M.I.T. Artificial Intell. Lab., Rep. AI-TR-720*, June 1983.