Dual Processing Spatially Distributed Integrating Fiber Optic Sensors for Non-intrusive Patient Monitoring

Xiaohua Xu

Project report 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
Physics Department

William Spillman, chair
Jimmy Ritter
Guy Indebetouw

April 21 2005
Blacksburg, Virginia

Keywords: perturbation, spatially integrating fiber optic, dual mode modulation sensing, non-intrusive patient monitoring

Dual Processing Spatially Distributed Integrating Fiber Optic Sensors for Non-intrusive Patient Monitoring

Xiaohua Xu

(Abstract)

Given the rapid aging of the world's population, improvements in technology for automation of patient care and documentation are badly needed. This project is based on previous research that demonstrated a 'smart' bed that can non-intrusively monitor a patient in bed and determine a patient's respiration, heart rate and movement without intrusive or restrictive medical measurements. The 'smart' bed is an application of spatially distributed integrating fiber optic sensors. The basic concept is that any patient movement that also moves an optical fiber within a specified area will produce a change in the optical signal. A statistical mode (STM) sensor and a high order mode excitation (HOME) sensor were previously investigated, based on which the author developed the present design including both modal modulation approaches. Development was made in both hardware and software for the combined STM/HOME sensor: a special lens system was installed allowing only the high order modes of the optical fiber to be excited and coupled into the sensor; computer-processing method was used for handling output from the dual STM-HOME sensor, which would offer comprehensive perturbation analysis for more reliable patient monitoring. Experimental results of simulating human body breathing and heartbeats by periodic mechanical perturbations are also presented, and the relative advantage and drawbacks of the two modal modulation approaches are discussed.

Acknowledgments

I wish to thank Dr. Jimmy Ritter and Dr. William Spillman, Jr. for offering me this invaluable research experience in Virginia Tech Applied Biosciences Center. Special thanks go to Dr. Spillman for providing me the opportunity of being involved in this project and related presentation in the 2004 Optics in the Southeast Conference (Raleigh-Durham, November, 2004). Thank him for being my advisor in the Applied & Industrial Master's Degree Program and for his priceless illuminating advice during the work. I also appreciate Dr. Guy Indebetouw for his role as a committee member and consultant of the project. Besides, I wish to thank Ken Meissner, Kai Chen for having been working together and everybody in Vtabc for their patience and guidance when I came to them with problems.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The continuing shortage of medical staff and the increase in the elder population due to the baby boom after World War II makes the automation of health care an ever-increasing priority. In particular, patient monitoring is very intrusive and labor intensive. Traditional medical measuring techniques may cause patient skin irritation or contribute to patient discomfort. In this work the improvement of a 'smart' bed to non-intrusively monitor patient respiration, heart rate and movement using spatially distributed integrating fiber optic sensors was carried out. These three parameters are extremely important in determining patient condition and preventing future problems for patients in nursing homes and extended care facilities. Measurement of respiration rate and heart rate provide immediate indication of whether a patient is in any distress, while the measurement of patient movement can be used to determine if any help is needed for patients with severely limited movement who are not able to change position conveniently to prevent occurrence of pressure sores. In clinical settings such as hospitals, outpatient surgery centers, or nursing homes, vital signs such as pulse and respiratory rates are measured by direct observation by skilled medical personnel. Continuous monitoring of vital signs requires attachment of sensors to the body in a number of ways. Monitoring of essential vital signs is an integral part of medical care. Pulse rate can be determined by placement of electrodes on the skin and monitoring of the electrocardiogram. The output of a fiber optic sensor placed on a finger, toe, or ear lobe and attached to a pulse oximeter can be used to determine pulse rate. Respiratory rate can be determined by chest movement as detected by changes in chest wall electrical impedance or inductance. Each method for detecting pulse or respiration requires an interface between the sensor and the patient's skin and the sensor must be held in place with an adhesive or by mechanical means such as Velcro. Any of these sensors can cause skin irritation or breakdown and may contribute to patient discomfort.

Pressure sores are a major cause of morbidity and mortality in the healthcare setting. As many as 1.5 million individuals are affected by pressure sores, at a total cost of 5 billion dollars annually. The prevalence of pressure sores in one US teaching hospital was 8%. Repositioning schedules are utilized as part of most preventive

measures in healthcare facilities. Recommendations are for repositioning bed-ridden patients every 2 hours and individuals in chairs at least once per hour. With digital signal collecting and computer data processing, a 'smart' bed can successfully provide continuous non-intrusive patient monitoring and record the information of the patient's movement that is valuable for the repositioning schedule. This is a new medical application of spatially distributed integrating fiber optic sensors.

1.2 History of Fiber Optic Sensors

It was in the year 1956 that Kapany first officially introduced the term "fiber optics". He defined fiber optics as the art of the active and passive guidance of light (rays and waveguide modes), in the ultraviolet, visible, and infrared regions of the spectrum, along transparent fibers through predetermined paths. Since then, fiber optics has been used in various photo-electronics devices, data processing, and photocopying systems. Today, optical fibers are primary components of global and local telecommunication systems. The primary method of transmitting information via optical fiber is digitally, that is, by sequences of optical pulses whose positions, widths, or occurrences can be modulated.

The invention of the laser in 1962 immediately stimulated a revolution that designers created fiber optic sensors combining optical fibers and optoelectronic devices with the application of this coherent light source. Fiber optic sensors are attractive because they offer excellent sensitivity and dynamic range, compatibility with optical data transmission and processing, long lifetimes and potential, low cost and high reliability. They are immune to electromagnetic interference. Conventional electrical sensors often require heavy shielding, significantly increasing cost, size, and weight. Fiber optic sensors can also work in other hostile environments, where there may be high temperatures, high-voltage, corrosive materials, all-solid-state configurations, vibration and explosion hazards, and where traditional sensors and transducers do not work well. The lightweight and small size of these devices are critical in such areas as aerospace and provide substantial advantages to many products. The comparable low cost is another advantage of fiber optic sensors. With each new successful product the cost of existing and newly introduced components continues to drop, opening the door for new waves of fiber sensor products. By 1990 the prices of components of fiber and light sources had fallen dramatically, and multiplexing elements were readily available at moderate cost. By the year 2000 the number of devices available at low cost had grown substantially, allowing fiber optic designers to produce a wide variety of devices

with superior performance at lower cost than existed technology while enabling the use of sensors in totally new areas of endeavor. These developments include the replacement of conventional spinning mass-inertial sensors with fiber optic gyros, widespread use of fiber optic sensors in process control and manufacturing, electrical isolation of patients in medicine, and fiber optic health monitoring systems in the aerospace and construction industries.

1.3 Fiber Optic Sensors Applied in a 'Smart' Bed

In this thesis, research on three kinds fiber optic sensors with two different techniques for sensing patient respiration rate and motion is discussed. Both techniques rely on modulation of the modal distribution in multimode optical fibers and can be implemented in a single sensor platform. The two modal modulation approaches were previously investigated [3], a statistical mode (STM) sensor and a high order mode excitation (HOME) sensor. STM focuses on different intensity distributions of the output resulting from inter-mode interference of all the guided modes of the fiber; HOME is based on the conversion of high order modes into low order modes. Since the dynamic ranges of the two techniques do not have much overlap, a third sensor combining them was studied. The present design includes an STM sensor combined with a HOME sensor, using both modal modulation approaches.

Generally, the author accomplished following tasks in this project. 1) A review of the literature on the development of fiber optic sensors and the fundamental theories of optical waveguides was first carried out. 2) A theoretical analysis was then presented concerning multi-mode wave propagation in optical waveguides based on both electromagnetic wave theory and geometric optics. 3) The combination of a statistical mode (STM) sensor and a high order mode excitation (HOME) sensor was modeled to provide a quantitative estimate of the sensing results against mechanical perturbations. 4) A high order mode generation lens system was developed for use with multimode optical fiber to realize HOME modal modulation. 5) Software and a user interface were developed to combine the STM and HOME approaches. 6) Experiments simulating human body breathing and heartbeats were carried out with the combined STM/HOME sensor with results presented and analyzed.

Chapter 2

Theoretical Background

2.1 Basic Principles for Optical Fibers

2.1.1 Wave Theory

Assume that the light in an optical fiber is coherent and linearly polarized and can be characterized as plane electromagnetic waves. The Maxwell's equations govern the behavior of time-varying electromagnetic fields.

$$\begin{aligned} \nabla \times \vec{E} &= -\frac{\partial \vec{B}}{\partial t} \\ \nabla \times \vec{H} &= \vec{J} + \frac{\partial \vec{D}}{\partial t} \\ \nabla \cdot \vec{B} &= 0 \\ \nabla \cdot \vec{D} &= \rho \end{aligned} \tag{2.1.1}$$

In the case of a charge-free, linear, homogeneous and isotropic medium, we take the curl of the first equation and substitute the second equation with J=0 to find Eq.2.1.2. According to the vector identity Eq.2.1.3, and $\nabla \cdot \vec{E} = 0$ (ρ=0), the steady state wave equation for the electric field is then derived (Eq.2.1.4). Similar procedure can be used to obtain the wave equation for the magnetic field (Eq.2.1.5). These two equations are also known as vector Helmholtz equations, whose solutions describe the propagation of energy in the medium. For the step-index waveguide the homogeneous wave equation can be solved in the core and cladding of the guide to obtain expressions for the fields.

$$\nabla \times \nabla \times \vec{E} = -\mu \frac{\partial}{\partial t} \nabla \times \vec{H} = -\mu\epsilon \frac{\partial^2 \vec{E}}{\partial t^2} \tag{2.1.2}$$

$$\nabla \times \nabla \times \vec{E} = \nabla\nabla \cdot \vec{E} - \nabla \cdot \nabla \vec{E} \tag{2.1.3}$$

$$\nabla^2 \vec{E} + k^2 \vec{E} = 0, \qquad k = -i\sqrt{\mu\epsilon}\frac{\partial}{\partial t} \tag{2.1.4}$$

$$\nabla^2 \vec{H} + k^2 \vec{H} = 0, \tag{2.1.5}$$

The solutions for the reflection and transmission of plane waves also take into account some boundary conditions. The most common type of boundary condition occurs when there are discontinuities in the dielectric constant. Such a discontinuity exists, for example at the core-cladding interface of a step-index fiber. For the dielectric media the general boundary conditions for both the normal and tangential components of the fields are

$$\begin{aligned} \hat{n} \times (\vec{E}_2 - \vec{E}_1) &= 0 \\ \hat{n} \times (\vec{H}_2 - \vec{H}_1) &= 0 \\ \hat{n} \cdot (\vec{B}_2 - \vec{B}_1) &= 0 \\ \hat{n} \cdot (\vec{D}_2 - \vec{D}_1) &= 0 \end{aligned} \tag{2.1.6}$$

Where n is the unit vector normal to a surface that forms the interface between two dielectric media.

Another important parameter that is a measure of the time average power flow in an electromagnetic field is the complex Poynting vector *S* (2.1.7). This is usually computed as the spatial intensity of the light passing through an optical fiber.

$$< \vec{S} >= \frac{1}{2} Re\{\vec{E} \times \vec{H}^*\} \tag{2.1.7}$$

2.1.2 Ray Optics

The standard approach to wave propagation analysis in optical fibers is wave theory. However, ray theory has an advantage in that the physical picture is clear and comprehensible. Ray optics describes the propagation of light fields by defining rays as the lines that cross the surfaces of constant phase of the light field at right angles. Ray optics can be applied to all phenomena that are described by the wave equation and that satisfy the additional requirement that the wavelength of light is short compared to the dimensions of the guide through which it passes. For example, ray optics can be used in a large-core multimode optical fiber.

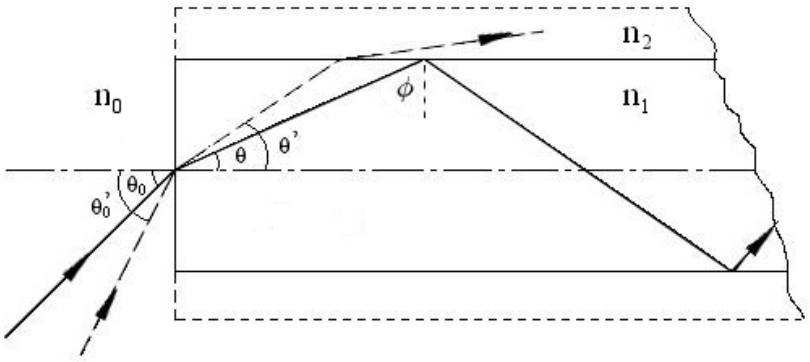

Fig2.1.1 Illustration of the ray passage along a fiber

An optical fiber is produced by forming concentric layers of cladding material with a low refractive index around a higher-index core region. Fig2.1.1 illustrates a flat-ended straight cylinder representing a uniform-core fiber. A ray incident at an angle θ_0 is refracted at the entrance end to an angle θ to the cylinder axis and strikes the wall of the cylinder at an angle $\phi = \pi/2 - \theta$. As long as the angle ϕ is equal to or greater than ϕ_c, this ray will be totally reflected and strike the other side of the cylinder at the same angle. Thus, this ray will be trapped inside the cylinder, undergo multiple total internal reflections, and emerge only at the other end of the cylinder. However, if a ray is incident at steeper angles such as θ_0', then such a ray, after refraction at the entrance end, is incident at an angle θ' to the axis of the cylinder and can strike the wall at an angle less than ϕ_c and thus refract out of the cylinder. Thus, it is clear that there is a limit to the acceptance angle of a fiber, which is set by the refractive index of the fiber core and the surrounding medium. This relationship between the acceptance angle and the refractive indices can be derived as follows. The critical angle ϕ_c is given by

$$\sin \phi_c = n_2/n_1 \tag{2.1.8}$$

Snell's law states that

$$\begin{aligned} n_0 \sin \theta_0 &= n_1 \sin \theta \\ n_0 \sin \theta_{NA} &= n_1 \sin[(\pi/2) - \phi_c] \\ &= n_1[1 - (n_2/n_1)^2]^{1/2} \end{aligned} \tag{2.1.9}$$

Therefore,

$$N.A. \equiv \sin\theta_{NA} = \frac{1}{n_0}\sqrt{n_1^2 - n_2^2} = \frac{n_1}{n_0}\sqrt{2\Delta} \tag{2.1.10}$$

This is an important quantity for multimode fibers -- the "numerical aperture". N.A. is defined as the sine of the maximum angle, with respect to the axis of the fiber, of an incident light ray that becomes totally confined within the fiber. Thus it determines a cone of acceptance angles for light that will be guided by the fiber. The larger the NA, the greater will be the amount of light accepted by the fiber. The N.A. of the fiber is directly related to the *normalized index difference.*

$$\Delta = \frac{n_1^2 - n_2^2}{2n_1^2} \tag{2.1.11}$$

2.2 Modes and Bending of Optical Fibers

2.2.1 Field Analysis

Mode is a complex mathematical and physical concept describing the propagation of electromagnetic waves. According to wave theory, a mode is an allowable field configuration, for a given waveguide geometry, that satisfies Maxwell's equations and all of the boundary conditions of the problem. From the view of ray theory, modes are simply the various paths light can take in a fiber. By *single-mode* we mean that there is only one path for the light; *multimode* means there are multiple paths. In a dielectric waveguide, the angle the ray takes with respect to the axis of the fiber can be related to the direction of the propagation vector associated with the mode. The ray along the axis is the lowest mode of the fiber. The ray that is reflected at high angles, the high-order mode, travels a greater distance than the low-angle rays to reach the end of the fiber.

There are an infinite number of mode with input angles within the N.A. of the fiber (Eq.2.1.10), however, the rigorous solution of Maxwell's equations [14] shows that for a given value of n_1 and n_2 and given fiber diameter only certain allowed modes can be guided inside the core. We discuss the problem here using cylindrical coordinates, with the fiber axis coincident with the z-axis. The radius r, the azimuthal angle ϕ, and the axial distance z are the three coordinates. Assume that the field solutions are linearly polarized in the fiber transverse plane: the electric field is polarized along x and the

magnetic field has y-polarization. Thus the wave equations Eq.2.1.4 and Eq.2.1.5 in cylindrical coordinates are expressed as

$$\frac{\partial^2 E_x}{\partial r^2} + \frac{1}{r}\frac{\partial E_x}{\partial r} + \frac{1}{r^2}\frac{\partial^2 E_x}{\partial \phi^2} + \beta_t^2 E_x = 0$$
$$\frac{\partial^2 H_y}{\partial r^2} + \frac{1}{r}\frac{\partial H_y}{\partial r} + \frac{1}{r^2}\frac{\partial^2 H_y}{\partial \phi^2} + \beta_t^2 H_y = 0 \tag{2.2.1}$$

β_t in the equation is the net transverse phase constant from $\beta_t^2 = \beta_r^2 + \beta_\phi^2$. β_t and the longitudinal propagation constant β satisfy the equation $\beta_t^2 + \beta^2 = n^2 k_0^2$, where nk_0 is equal to the wavevector magnitude and $k_0 = 2\pi/\lambda$ is the free-space phase constant. We assume that the solution for E_x or H_y is a discrete series of modes, each of which has separated dependences on r, ϕ, and z in product form:

$$E_x(or\ H_y) = \sum_i R_i(r)\Phi_i(\phi)\exp(-i\beta_i z) \tag{2.2.2}$$

Substituting Eq.2.2.2 into Eq.2.2.1, we find that the separation of variables is possible and Eq.2.2.1 yields

$$\frac{d^2\Phi}{d\phi^2} + \ell^2\Phi = 0$$
$$\frac{d^2R}{dr^2} + \frac{1}{r}\frac{dR}{dr} + \left[\beta_t^2 - \frac{\ell^2}{r^2}\right]R = 0 \tag{2.2.3}$$

The second equation is known as Bessel's differential equation. The general solutions of these two equations are

$$\Phi(\phi) = \begin{cases}\cos(\ell\phi+\alpha)\\ \sin(\ell\phi+\alpha)\end{cases}$$
$$R(r) = \begin{cases} AJ_\ell(\beta_t r)+A'N_\ell(\beta_t r) & \beta_t\ real\\ CK_\ell(|\beta_t|r)+C'I_\ell(|\beta_t|r) & \beta_t\ imaginary\end{cases} \tag{2.2.4}$$

where α is a constant phase shift and the quantity *l* is the angular or azimuthal mode number for LP (linearly polarized) modes, which must be an integer. A, A', C, and C' are arbitrary constants; J_l, N_l, denote the *lth* order ordinary Bessel functions of the first and second kinds; and K_l, I_l are the *lth* order modified Bessel functions of the first and second kinds, respectively. Field solutions are first obtained independently for the core

and cladding regions and are connected later at the core-cladding boundary so as to satisfy boundary conditions. However, two parameters should be given for each mode throughout the two regions: azimuthal mode number l and phase constantβ_t. Besides, the basic properties of a guided mode in the fiber require that (1) the solution in the core is oscillatory, exhibiting no singularities, and (2) the solution in the cladding monotonically decreases as radius increases. Therefore the ordinary Bessel functions should apply in the core region, where $\beta_{t1}^2 = n_1^2 k_0^2 - \beta^2 > 0$ and the electromagnetic field is oscillating. So β_{t1} is real. Also, only the J_l functions can be used because the N_l functions are infinite at r=0. The second condition indicates that the modified Bessel functions and only K_l functions can be used in the cladding with $\beta_{t2}^2 = n_2^2 k_0^2 - \beta^2 < 0$, to show that the electromagnetic field is evanescent with an exponential-like behavior. So β_{t2} is required to be imaginary in the cladding region. The solution of the equations is thus

$$R(r) = \begin{cases} AJ_\ell(ur/a) & r \leq a \\ CK_\ell(wr/a) & r \geq a \end{cases} \tag{2.2.5}$$

Here we define the normalized transverse phase and attenuation constants, respectively, as

$$\begin{aligned} u &= \beta_{t1} a = a(n_1^2 k_0^2 - \beta^2)^{1/2} \\ w &= |\beta_{t2}| a = a(\beta^2 - n_2^2 k_0^2)^{1/2} \end{aligned} \tag{2.2.6}$$

where the refractive indices in the core and the cladding and core diameter are denoted by n_1, n_2, and a, respectively, and $k_0 = 2\pi/\lambda$ is the free-space phase constant. We may also define a dimensionless parameter V called the normalized frequency, as it depends only on the characteristics of the guide and the light wavelength (or frequency):

$$V = (u^2 + w^2)^{1/2} = ak_0(n_1^2 - n_2^2)^{1/2} = n_1 ak_0\sqrt{2\Delta} \tag{2.2.7}$$

The V number is often called modal number. It suggests that only a finite number of modes are allowed in propagation with given refractive indices of the fiber core and cladding and the fiber diameter. Taking into account the continuity of the field components tangential at the dielectric discontinuities located at r=a, the relationship between u and w is obtained for even modes and odd modes differently. See equations in fig.2.2.1. It shows that the guiding structure can support only discrete modes whose characteristics can be found by a graphical method illustrated in the figure. The solutions of u and w for the various modes are found from the intersections of circle of radius V and of the curves representative of the equations.

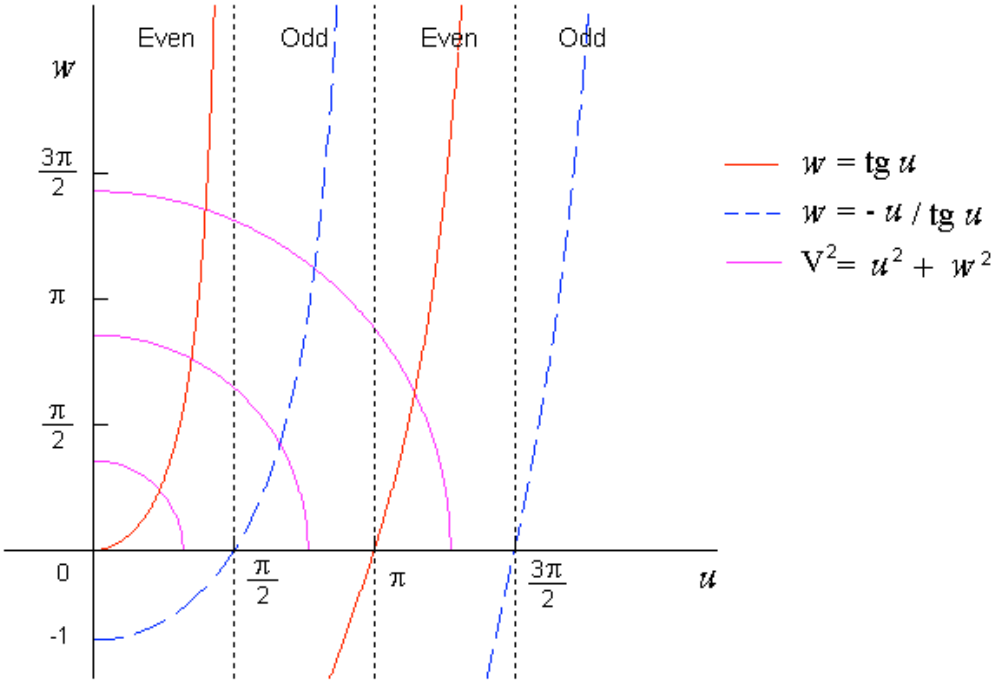

Fig2.2.1 Graphical solution for determining the parameters *u* and *w*

2.2.2 Modes Conversion and Bends

Fiber optics makes use of the property of fibers to conduct light along curved paths. Three factors contribute to the propagation characteristics of a bent waveguide: pure radiation losses, transition losses between the straight and the bent waveguide, and the phase constant of the propagating field. In practice the radiation losses are ultimately determined by several practical aspects such as the roughness of the waveguide and the technological process itself. The waveguide is then always used under the influence of the two remaining factors— the coupling loss and the phase constant.

The standard electric field of a plane wave with the direction of propagation along the z-axis and the electric field oriented along the x-axis is shown in Eq.2.2.8. The mode phase constant β is equivalent to the longitudinal propagation constant, or the z component of the wavevectors of the constituent plane waves: $\beta = k \cos \theta$.

$$E = E_0 e^{-i\vec{k}\cdot\vec{r}} = E_0 e^{-ik_x x} e^{-i\beta z} \tag{2.2.8}$$

Once the modes of the straight waveguide are determined, the mode shape distortion and phase constant perturbation for each desired bending radius could be calculated by the expansion with the modes of the straight waveguide and the first horizontal leaky mode under the cutoff. This method is valid for every kind of dielectric waveguide, single mode as well as multimode. [4]

$$\beta = \beta_1 + \beta_1 \frac{c_{11}}{R} + \frac{\beta_1 \beta_2 c_{12}^2}{(\beta_1 - \beta_2) R^2} + \cdots \tag{2.2.9}$$

Seen from Eq.2.2.9, for very large bending radius R, the mode phase constant β tends to that of the straight waveguide fundamental mode β_1. For sharp bends β increases with the decreasing of the bending radius R, consequently resulting in a smaller θ to the axis. Clearly, the phase constant perturbation contributes a modes conversion from high-order modes to low-order modes with the relationship of "perturbation ~ 1/R ~ β ~ cos θ".

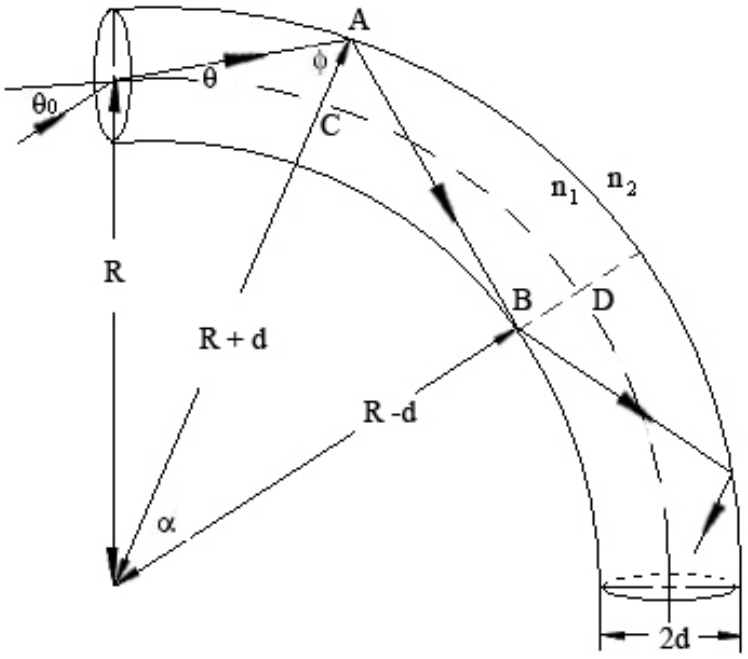

Fig2.2.2 Ray passing along a bent fiber

Modes can also be considered from the standpoint of geometric optics. Fig2.2.2 shows a curved fiber of diameter 2*d* bent with radius R (from center of the arc to axis of the fiber). The incident ray at an angle θ_0 is refracted at the end of the fiber at an angle θ, and the incident angle at outside fiber-air boundary is ϕ with respect to the normal. This angle ϕ is given by the following expression:

$$\sin\phi = \frac{\cos\theta(R+h)}{R+d} \tag{2.2.10}$$

where *h* is the incident height of the ray with respect to the fiber axis. The path length between two successive internal reflections AB = *l* is given by

$$l = \frac{(R-d)\sin\alpha}{\sin\phi} \tag{2.2.11}$$

This intercept length of the ray is the same between any pair of successive reflections. It can be shown that, the reflection path in the curved fiber *l* is related to the axial length between CD = *L*, as follows:

$$\frac{l}{L} = \frac{l}{R\alpha} < \frac{l}{(R-d)\alpha} = \frac{(\sin\alpha)/\alpha}{\sin\phi}$$

$$l_{tot} < \frac{L_0}{\sin\phi} \tag{2.2.12}$$

where l_{tot} is the sum of all the successive intercept lengths and L_0 is the axial length of the whole fiber. Now, $(\sin\alpha)/\alpha$ is less than or equal to unity, so that the inequality becomes Ineq.2.2.12. Therefore, the total length of the reflection path for a ray in a fiber bent over a circular arc is less than that of the same ray in a straight fiber; that is, ray travels in a lower-order mode path of shorter distance when the fiber is curved than when it is straight.

2.3 Antenna Gain in Integrating Fiber Optic Sensors

Integrating fiber optic sensors offer the potential to monitor large spatial extents due to their geometric flexibility. With the increasing of the length of an integrating fiber optic sensor, its antenna gain or gauge length can be increased. Long gauge length

sensors have the ability of scaling its sensitivity in regions where the parameter of interest varies spatially, suppress unwanted modal signals, and thus serve as 'matched filters' for the spatial distribution of that parameter field of interest. [5]

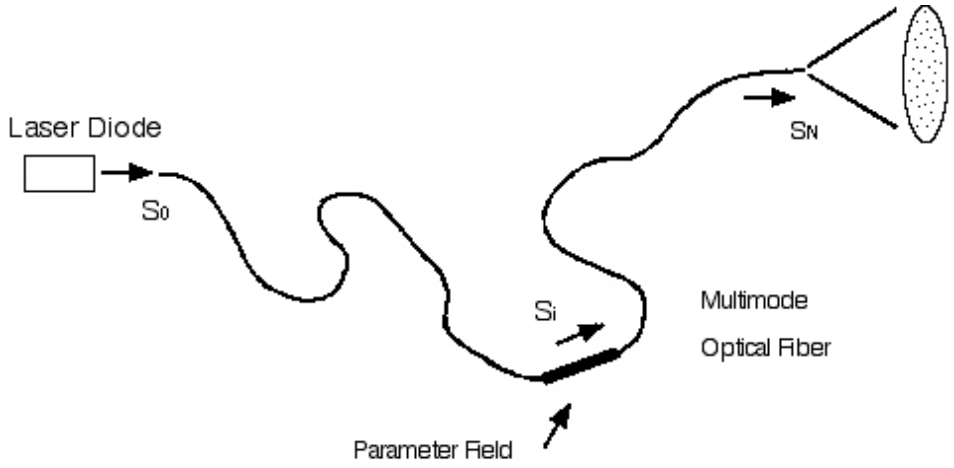

Fig.2.3.1 Schematic diagram of a long gauge length sensor

Fig.2.3.1 depicts a generalized long gauge length spatially distributed fiber optic sensor, which is assumed to consist of N segments and the i*th* segment is immersed in the i*th* parameter field of interest. For a phase modulating sensor, the key parameter of the signal is its phase φ, and the input signal from one end of the fiber can be simplistically represented as

$$s_0 = e^{i\varphi_0} \tag{2.3.1}$$

While each segment is considered as independent with its own operator to transform the signal into the next segment, so the final output signal would be

$$s_N = e^{i\varphi_N} = \left[\prod_{i=1}^{N} e^{i\varphi_i}\right] e^{i\varphi_0} \tag{2.3.2}$$

$$\varphi_N = \sum_{i=1}^{N} \varphi_i(t) + \varphi_0 = mP(t) + \varphi_0 \tag{2.3.3}$$

As is discussed before, the phase propagation constant can be treated as a term proportional to the perturbation (or the inverse of the bent radius) plus a constant term. It is also assumed that the only m fiber segments subjected to the same perturbation *P(t)*

are co-located, and no other segments modulate the phase. Therefore the change of the output pattern from the fiber would be the absolute value of the first time derivative of the light intensity, which is a function of the phase φ. See Eq.2.3.4.

$$\frac{d}{dt}\{I(\varphi)\} = \frac{\partial I}{\partial \varphi} \cdot m \cdot \left\{ \frac{dP(t)}{dt} \right\}$$

$$\left| \frac{dI}{dt} \right| \propto m \left| \frac{dP(t)}{dt} \right| \tag{2.3.4}$$

This result is particularly the output from an STM (statistical mode sensor), which scales in amplitude with the number of times the gauge length is multiplied in perturbation regions, and with the derivative of the detected field. It indicates that, despite the antenna configuration, the antenna gain of long gauge length sensors will provide selectivity for detecting parameter distributions of interest such as the perturbation to the signal phase. This filtering ability can be extremely useful in civil structure applications in that the sensor antenna is functioning as an analog computer providing a preprocessing and data reduction function. If a known potential damage condition can be identified in terms of particular modal shape distributions, a long gauge length sensor can be configured to select for that particular modal shape, providing a simple digital output. When critical levels are exceeded, a warning could be provided. The simplicity and potential low cost of such long gauge length fiber optic sensors make them very attractive for sensing applications where coverage of a large spatial extent is required.

2.4 Principles of Sensors for a 'Smart' Bed

2.4.1 General Concepts

When a multimode optical fiber is excited by a coherent optical source, the output projected on a screen is seen to be a granular pattern that consists of numerous speckles. This speckle pattern is observed to change once a perturbation towards the fiber affects the angular propagation of the light. With the intensity of the light inside the fiber redistributed, some speckles become brighter, some become darker and some remain unchanged. However, the total intensity of all the speckles remains constant.

$$I_T = \sum_{k=1}^{N} I_k = const \tag{2.4.1}$$

where I_k is the individual speckle intensity, N is the number of speckles.

The expression for the intensity of each speckle I_k can be obtained by integrating the spatial intensity of the light in the optical fiber over the area of the *kth* speckle [7], shown as Eq.2.4.2. A_k, B_k, ϕ_k and δ_k are constant values for any given k. A_k accounts for the mode self-interaction and B_k for the steady stated mode-mode interaction. The third term is another mode-mode interaction term with a modification of ϕ_k and the perturbation $P(t)$.

$$I_k = \int_{a_k} I da_k = A_k \left\{1 + B_k \left[\cos(\delta_k) - P(t)\phi_k \sin(\delta_k)\right]\right\} \tag{2.4.2}$$

Thus, the total intensity I_T can be obtained by the sum of each I_k,

$$I_T = \sum_{k=1}^{N} A_k + \sum_{k=1}^{N} A_k B_k \cos(\delta_k) - \sum_{k=1}^{N} A_k B_k P(t)\phi_k \sin(\delta_k) \tag{2.4.3}$$

Assume that the first two terms change much more slowly than the perturbation related term, so they can be filtered out during the signal processing in a way that only the time varying term is left representing the perturbation of the fiber.

$$\Delta I_T = \Delta\left[-\sum_{k=1}^{N} A_k B_k P(t)\phi_k \sin(\delta_k)\right]$$

$$\Delta I_T = \left[-\sum_{k=1}^{N} A_k B_k \phi_k \sin(\delta_k)\right]\Delta P(t) \tag{2.4.4}$$

From Eq.2.4.1, the sum of the intensities of all the speckles remains unchanged, which makes Eq.2.4.4 approach zero for a large number of N. Therefore, either taking the absolute value of the changes or summing a small number of intensities should be considered as a modification to carry out non-zero experimental outputs, which are the fundamental ideas for STM sensor and HOME sensor respectively, if the mode volume of the fiber is not completely filled in the absence of perturbation.

2.4.2 STM Sensor

In a STM sensor, the sum of the absolute values of the changes in all the signals is taken; that is, the intensity of each speckle is compared with itself over a short time interval Δt and the summation of the absolute value of all the differences makes the total change ΔI_T. We take the absolute value and the first time derivative of both sides of Eqs.2.4.4, then, since Δt is constant during the experiment, as the frame sampling interval of the digital camera, we can put Δt to the r.h.s.

$$\left|\frac{\Delta I_T}{\Delta t}\right| = \left[\sum_{k=1}^{N}\left|-A_k B_k \phi_k \sin(\delta_k)\right|\right]\left|\frac{dP(t)}{dt}\right|$$

$$\left|\Delta I_T\right| = \sum_{k=1}^{N}\left|I_k(t+\Delta t) - I_k(t)\right| = \left|\Delta t\right|\left[\sum_{k=1}^{N}\left|-A_k B_k \phi_k \sin(\delta_k)\right|\right]\left|\frac{dP(t)}{dt}\right| \tag{2.4.5}$$

As for a large number of components, the terms summed within the brackets remains a constant value. Besides, the output refers to the average value of the intensity change between two data collections, which adds the term Δt/2, thus the final expression of the output of an STM sensor is written as:

$$STM\left(t+\frac{\Delta t}{2}\right) = \sum_{k=0}^{N*M-1}\left|I_k\left(t+\Delta t\right) - I_k(t)\right| = C\left|\frac{dP\left(t+\Delta t/2\right)}{dt}\right| \tag{2.4.6}$$

Based on Eq.2.4.6, the pixel-to-pixel intensity differences between every two successive frames from a digital camera are calculated. Seen from Fig.2.4.1, as an STM sensor works, the output pattern of the optical fiber is projected on the screen of a digital camera and frames are taken every Δt seconds. During the signal processing, a frame of data of the speckle information is recorded and stored in memory and a second frame is recorded afterwards. The change in each individual pixel between the first and the second frame is then calculated and the sum of the absolute values of all changes is obtained and outputted. The second frame then replaces the first frame in memory, a new frame is recorded and the process is repeated. The flow chart of the process is shown in Fig.2.4.2.

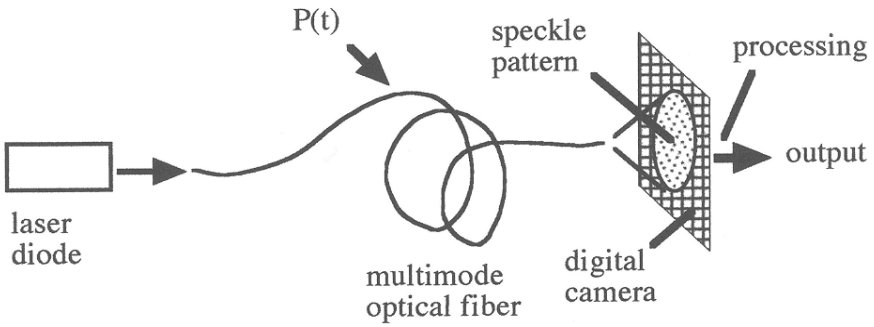

Fig.2.4.1 STM Sensor Schematic Diagram

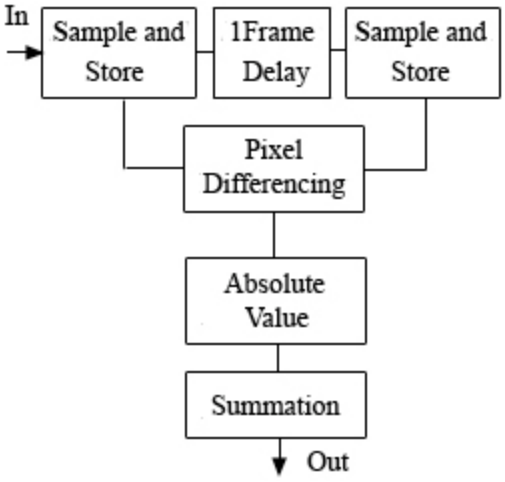

Fig.2.4.2 Flow Chart of STM Sensor Data Processing

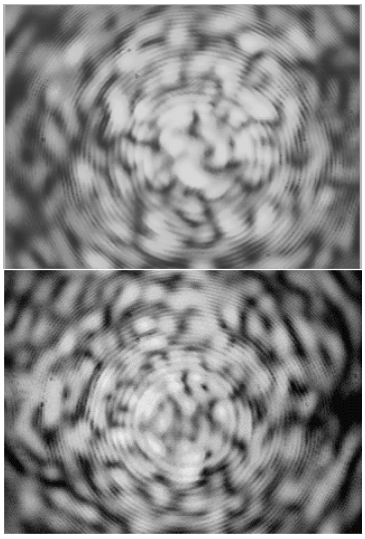

Fig.2.4.3 Speckle Patterns

2.4.3 HOME Sensor

In the input of a HOME sensor, only the high order modes are excited, which results in an annulus-shaped output when projected on a screen, as is seen in Fig.2.4.4, A large area detector is positioned at the interior of the annular output, where there is no light going through or getting intercepted until the fiber is perturbed. We know from former discussion that the higher order modes of the fiber will convert into the lower

order modes in the presence of the bending of the fiber caused by a perturbation. Therefore, the detector can measure the amount of the mode conversion from the high order modes to the low order mode that is directly related to the magnitude of the perturbation *P(t)*.

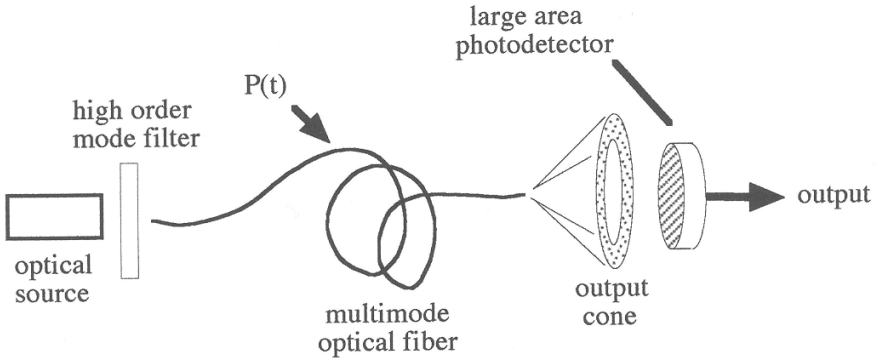

Fig.2.4.4 HOME Sensor Schematic Diagram

Projecting the output from a HOME sensor fiber on a screen of N pixels, we would obtain an annulus speckle pattern with pixels 1 to M illuminated and pixels M+1 to N in the dark. As mentioned in chap 2.4.1 that Eq.2.4.4 approaches zero for a large number of N so that the total light intensity of all the N pixels on the screen keeps constant.

$$I_{tot} = \sum_{i=1}^{M} I_i + \sum_{j=M+1}^{N} I_j = Const \tag{2.4.7}$$

After the perturbation *P(t)* is applied, some pixels that were originally bright become dim, and some pixels get illuminated due to the perturbation. The changes of light intensities ΔI_i for pixels 1 to M are summed up in Eq.2.4.8. The total amount of optical power loss from these initially illuminated pixels is directly proportional to the magnitude of the perturbation *P(t)*. Similarly, the light intensity changes ΔI_j for pixels M+1 to N provides the amount of optical power gained by those initially dark pixels.

$$\sum_{i=1}^{M} \Delta I_i = -\sum_{i=1}^{M} \alpha_i P(t) \tag{2.4.8}$$

$$\sum_{j=M+1}^{N} \Delta I_j = \sum_{j=M+1}^{N} \beta_j P(t) \tag{2.4.9}$$

It is evident that Eq.2.4.8 and Eq2.4.9 add up to zero because the total optical power loss is equal to the total optical power gain on the screen; either one of them gives a non-zero result that is proportional to the perturbation. In particular, Eq.2.4.9 refers to light intensity changes of the area where the detector is placed. Thus the amount of the mode change from high-order-mode to low-order-mode is expected to be directly related to the perturbation.

$$\Delta I_{tot} = \sum_{i=1}^{M} \Delta I_i + \sum_{j=M+1}^{N} \Delta I_j = \left(\sum_{j=M+1}^{N} \beta_j - \sum_{i=1}^{M} \alpha_i \right) P(t) = 0 \tag{2.4.10}$$

The output expression of a HOME sensor is therefore

$$HOME(t) = \left(\sum_{j=M+1}^{N} \beta_j \right) P(t) \tag{2.4.11}$$

2.4.3 Combined STM/HOME Sensor

The combination of an STM sensor and a HOME sensor takes into account both of the two modal modulation approaches: (1) only the high order modes of the optical fiber are excited by a *coherent optical source*; (2) the output is projected onto a digital camera with N*M pixels (320*240 pixels in the experiment) as in an STM sensor. As is shown in the schematic diagram in Fig.2.4.5, a bright annulus is observed on the screen when there is no perturbation applied to the fiber. As soon as the optical fiber is perturbed, the annular speckle pattern will twist and expand into the central area of the screen where it was dark before, which indicates a mode conversion from high order modes of the fiber into low order modes. The combined STM/HOME sensor, however, measures the low order modes of the fiber by summing the intensities on the pixels in the middle of the screen, instead of the measurement with an integrating detector in the original HOME sensor.

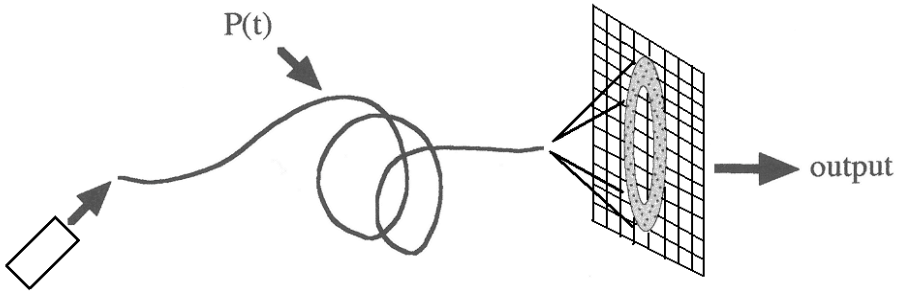

Fig.2.4.5 Combined STM/HOME Sensor Schematic Diagram

For a combined STM/HOME sensor, two resultant outputs are obtained with respect to the two different sensing methods. One is the summation of all the absolute values of the pixel-to-pixel changes in light intensity on the screen between every two time frames, including all the N*M pixels with x-coordinate from 0 to N-1 and y-coordinate from 0 to M-1. See Eq.2.4.12. The other result is the sum of light intensities on pixels of the central area on the camera screen, as is given in Eq.2.4.13.

$$COMB1\left(t+\frac{\Delta t}{2}\right)=\sum_{k=0}^{N*M-1}\left|I_k(t+\Delta t)-I_k(t)\right|=\left|\frac{dP(t+\Delta t/2)}{dt}\right| \tag{2.4.12}$$

$$COMB2(t)=\sum_{i,j}I(i,j) \qquad where \quad (i-x_0)^2+(j-y_0)^2<R^2 \tag{2.4.13}$$

R is the inner radius of the annulus, (x_0, y_0) *is the center.*

With both of the experimental outputs, this combined modification of STM sensor and HOME sensor has an advantage of offering more comprehensive analysis on the information of perturbation *P(t)*, particularly in that it results in an increased dynamic range – when the high sensitivity STM approach saturates due to large perturbation, the low sensitivity HOME approach becomes active.

Chapter 3

Experimental Improvements

3.1 System Description

The Combined STM/HOME sensor system includes two main parts, a fiber optic transducer and a data transmission processing element.

(1) Fig.3.1.1 shows the optical experimental set up for high order modes inputs to the fiber optics sensor. A linearly polarized, coherent laser source of 75mW at 632.8nm wavelength was used to excite a step-index silica multimode optical fiber with 400 μm core size. An additional experimental set up for generating homogeneously distributed annular patterns is shown in Fig.3.1.2 (More discussion in section 3.2). It consists of two lenses, one opaque disk and an iris diaphragm with a maximum aperture diameter of 41.3mm and a minimum of 2.4mm. Adjusting the diameter of the iris can change the thickness of the annulus patterns.

(2) The second part involves the data transferring and processing section. When optical signals are collected by a CCD camera that is placed at one end of the optical fiber, the data is transferred into the buffer of a wireless transmitter once every Δt time interval. The data is then wirelessly transmitted to a receiver that is connected to a laptop computer via a USB interface. Afterwards, a self-designed Visual C++ program carries out the data processing. As is seen in Fig.3.1.3, the black box beside the laptop is the remote module with a built-in digital camera and a wireless transmitter for data transmission.

(3) The algorithm for the Combined STM/HOME sensor system is written in Visual C++6.0. Microsoft AVICap class is adopted in the program to control the external video source. The main procedures are placed in the file "SpeckleDlg.cpp". The whole program has a parallel structure as shown in Table 3.1. The SpeckleDlg.cpp file is attached as Appendix.

Fig.3.1.1 Optical Experimental Set Up for High Order Modes Inputs

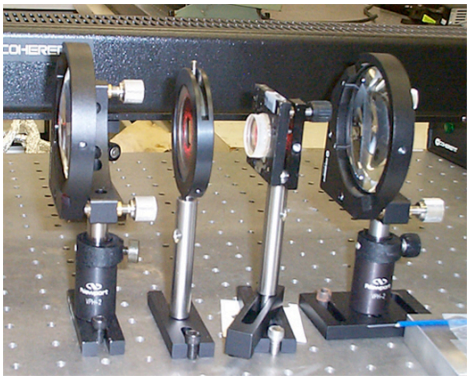

Fig.3.1.2 Additional Setup

Fig.3.1.3 Data Transferring and Processing Section

<table>
<tr><td colspan="2">1. Start Operating Window
CspeckleDlg::OnInitDialog():
1.1 Find the CCD camera and initialize parameters
1.2 Assign screen area for displaying CCD frames
1.3 Set frame callback function
1.3 Enable timer with 100ms interval for drawing graphs</td></tr>
<tr><td rowspan="2">2. Data Transferring and Processing
CALLBACK FrameCallbackProc():
2.1 Increase frame number
2.2 Read data from video source to buffer bFrame[0] and bFrame[1] alternatively for successive frames
2.3 Calculate current COMB1 value and save to array SPD[]
2.4 Calculate current COMB2 value and save to array iarea_c[]
2.5 Back to waiting loop</td><td>4. Select Display Scale
CSpeckleDlg::OnCheck
4.1 Display graph with auto-scale (or relative scale)
CSpeckleDlg::OnVScroll
4.2 Change graph’s absolute scale
4.3 Back to waiting loop</td></tr>
<tr><td>5. Click Button Format
CspeckleDlg::OnFormat():
5.1 Call capDlgVideoFormat() to select video format
5.2 Back to waiting loop</td></tr>
<tr><td rowspan="2">3. Timer Operation
CspeckleDlg::OnTimer():
3.1 Send Message to activate OnPaint()
CspeckleDlg::OnPaint()
3.2 Redraw prior curves with background color for “erasing”
3.3 Draw new curves of recent COMB1 and COMB2 values with red color
3.3 Back to waiting loop</td><td>6. Click Button Source
CspeckleDlg::OnSource():
6.1 Call capDlgVideoSource() to select video source and its options
6.2 Back to waiting loop</td></tr>
<tr><td>7. CspeckleDlg::OnSave():
7.1 pop up “Save As” window to specify directory and filename for data saving
7.2 Save array SPD[] and iarea_c[] to specified file
7.3 Back to waiting loop</td></tr>
<tr><td colspan="2">8. Click Button Quit
CspeckleDlg::OnOK():
8.1 Disconnect video source
8.2 Quit program</td></tr>
</table>

Table 3.1 Program Structure

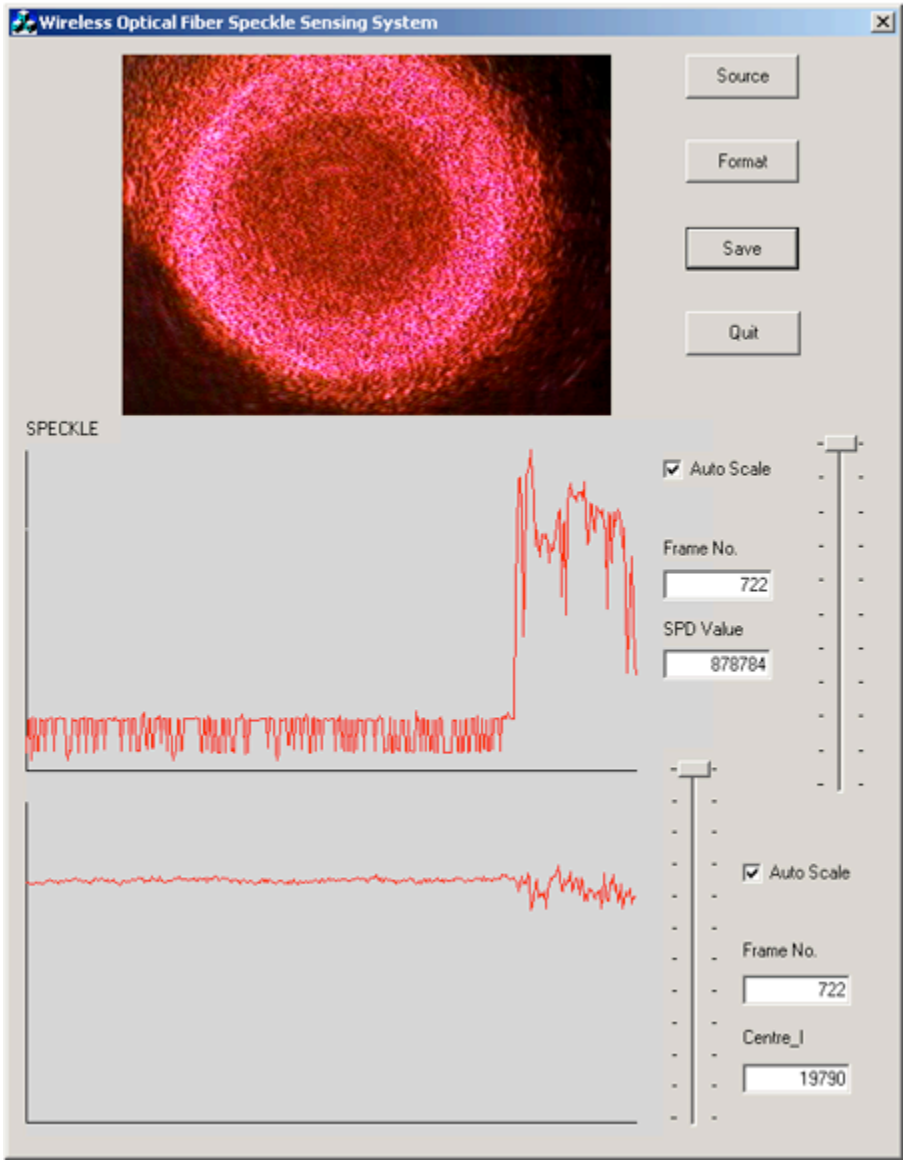

Fig.3.1.4 Operating Window

3.2 High Order Mode Generation

In order to realize a combined STM/HOME sensor, high order modes must be injected into the multimode optical fiber used. Two different methods were tested. In the first one, the incident laser light was injected into the fiber at an angle instead of along the axis of the fiber. See Fig.2.4.5. The result turns out to be unsatisfactory as can be seen in Fig.3.2.1. The light is asymmetrically distributed on the screen due to the asymmetry of the incident light. This leads to an error in the center point calculation – the center point calculated is (163, 114) in screen pixel dimensions (array = 320*240). Fig.3.2.2 below shows an in-line method to solve the problems in the "angle method".

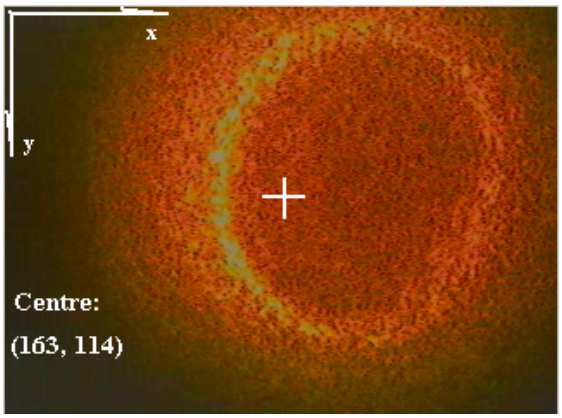

Fig.3.2.1.Annulus Output from the "angle method"

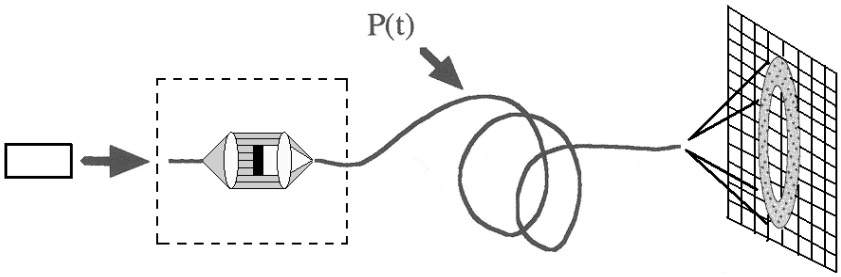

Fig.3.2.2. Schematic Diagram for Modified Combined STM/HOME Sensor

The improved high order mode generation method introduces one simple additional experimental set up: two lenses are placed parallel to each other and an opaque disk is placed between them to block the light from central area. A completely in-line version would consist of fiber, lens, opaque disk, lens fiber. For this experiment, the laser was placed directly at the focus of the first lens, replacing the input fiber shown in Fig.3.2.3. Homogeneously distributed annular patterns were generated. The resultant output greatly reduces the calculation error in data processing (detailed in chapter 3.3).

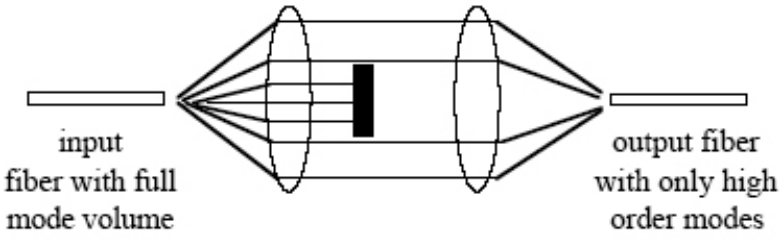

Fig.3.2.3 Bulk Optic High Order Mode Generator

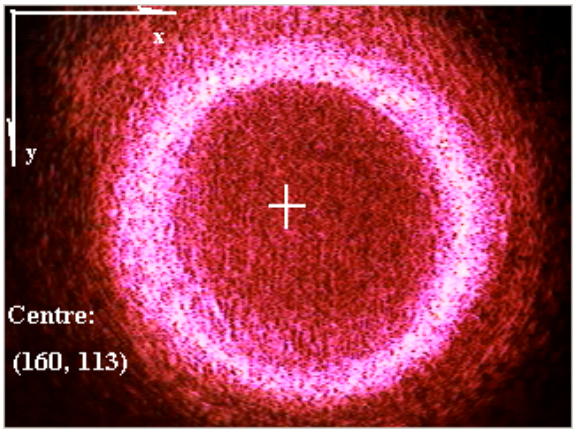

Fig.3.2.4.Annulus Output from Substituted Method

3.3 Determination of the Center Point of Intensity

For the sake of accuracy, digital analysis was used to locate the centre point of the circle. As all the intensity information of every pixel on the camera screen is stored in computer memory, the centre of the annulus pattern could be determined by calculating the weighted mean value of all the coordinator values (x_i, y_i) of pixels with their respective intensity values I_i as the weight, thus the annulus centre is selected as the average point of the total light intensity distribution on the screen.

$$x_c = \left(\sum_{i=1}^{n} I_i x_i\right) / \left(\sum_{i=1}^{n} I_i\right)$$

$$y_c = \left(\sum_{i=1}^{n} I_i y_i\right) / \left(\sum_{i=1}^{n} I_i\right) \tag{3.3}$$

Due to noise, the calculated annulus center was not exactly the geometric center of the output pattern as determined by visual inspection. A modification was then used to reduce the data and obtain a more accurate result. Fig.3.3.1 shows the intensity distribution along the horizontal line across (x_c, y_c), the center point of intensity calculated by weighted mean value method. Peaks are shown where the corresponding illuminated annulus pattern is located on the screen, but we are unable to identify the middle point between the two peaks from this graph. After taking a threshold value of 200 and setting all the intensity data lower than 200 into zero, the graph is greatly simplified. As is seen in Fig.3.3.2, the first two points from the middle with non-zero modified values define the edges of the annulus, and the x-coordinate of the center point is determined to be at the 158th pixel out of 320 pixels in the horizontal line. The same process is applied to obtain the vertical center point of intensity. Clearly, the center point calculated by the modified method matches the geometric center of the screen obtained by visual inspection.

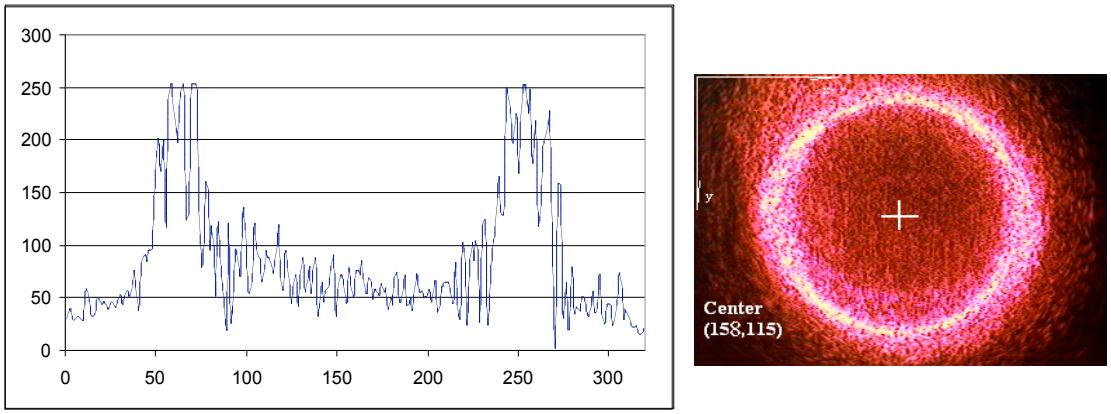

Fig.3.3.1 Determination of Center Point with Unmodified Data

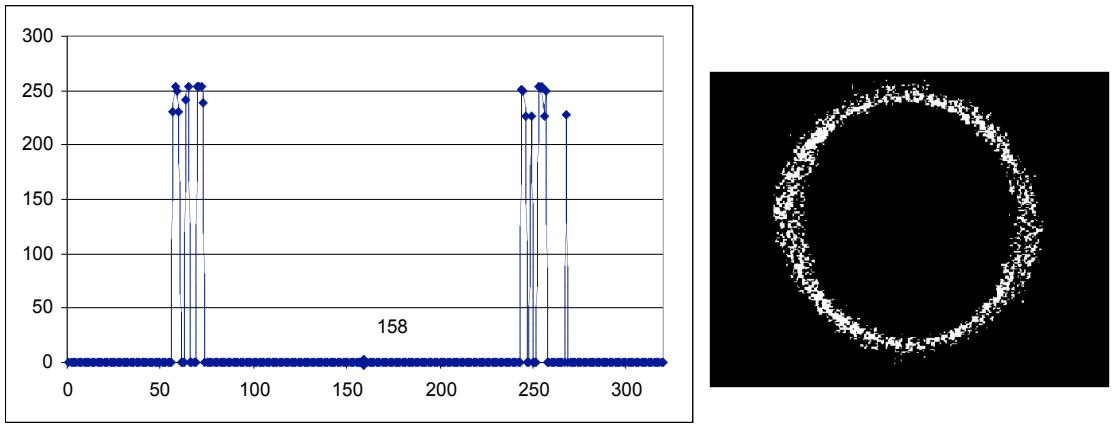

Fig.3.3.2 Determination of Center Point with Modified Data

Chapter 4

Results and Discussion

4.1 Quantitative Tests

In order to evaluate the capability of a combined STM/HOME sensor to determine a patient's respiration, heart rate and movement, two kinds of quantitative tests were carried out. In the first, a periodic perturbation from a pendulum was utilized to simulate patient heartbeats with the average pulse rate of 70 cycles/minute (frequency 1.167 Hz). As is shown in Fig.4.1.1, the optical fiber was attached to the bottom of the pendulum and was swinging at a small amplitude compared to the length of the pendulum, so that the movement could be considered as sinusoidal oscillation. The period of the oscillation could be obtained by the calculation of the period of an ideal pendulum: $T = 2\pi\sqrt{L/g}$. To simulate a perturbation from the heartbeats of human being, the period in this experiment was taken approximately as 0.856 s when the pendulum length was set as 18.2 cm.

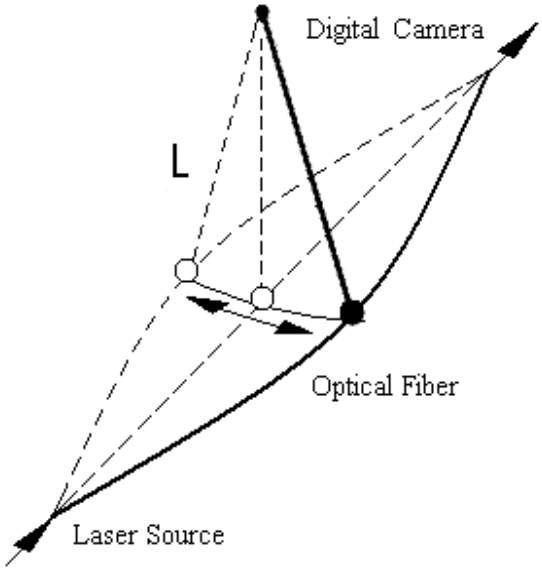

Fig.4.1.1 Schematic Diagram for Perturbation Test

We have discussed in former chapter that the COMB1 (STM) output is equal to the absolute value of the first time derivative of the perturbation *P(t)*, while the COMB2 (HOME) output is directly proportional to the perturbation. Now that the oscillation of the pendulum is approximated as sinusoidal, the experimental output from a combined STM/HOME sensor is expected to be

$$COMB1 \propto \left| \frac{dP(t)}{dt} \right| \approx |\cos\theta| \tag{4.1.1}$$

$$COMB2 \propto P(t) \approx \sin\theta \tag{4.1.2}$$

The equations indicate that the power spectrum of the COMB1 output should be at twice the frequency of the power spectrum of the COMB2 output due to the absolute value taken during COMB1 processing. The experimental results show a good agreement with the actual perturbation frequency. In the two power spectra in Fig.4.1.2, a peak is clearly shown at 1.17 Hz for COMB2 output and for COMB1 output one peak is located at doubled frequency 2.34 Hz.

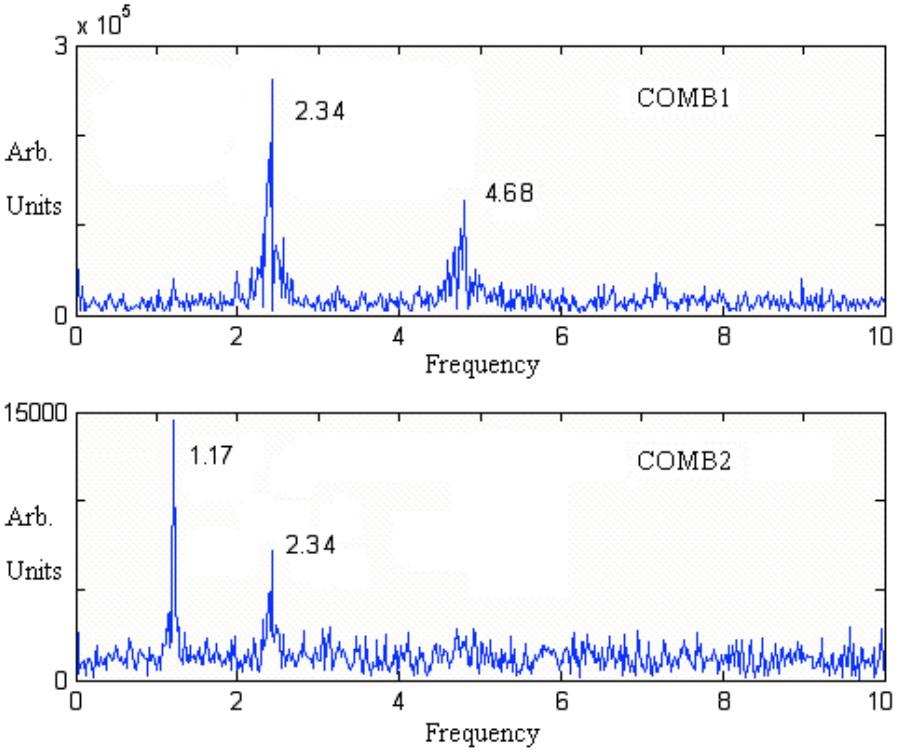

Fig.4.1.2 Simulation at an Approximate Human Heartbeat Rate

Another simulation, at the human body's breathing rate of ~ 9 cycles/minute (frequency 0.15Hz), was made using manual periodic disturbances. As a result, the power spectrums of COMB1 and COMB2 have peaks at 0.15 Hz and 0.3 Hz respectively (Fig.4.1.2). One can also note that the COMB1 signals are both larger than the COMB2 signals by a factor of 15~20. This implies that for small perturbations, there would be a better signal to noise ratio in the COMB1 output than the COMB2 output. However, for large perturbations (rapid strained breathing, gross patient motion) the COMB1 signal will saturate, i.e. phase differences in the speckle pattern will exceed 2π and the signal to noise ratio will deteriorate significantly. The COMB2 signal, on the other hand, will become more effective at representing the actual perturbation signal. This points out the major advantage of the combined sensor: it significantly extends the dynamic range over each sensing technique in isolation without sacrificing the benefits of either.

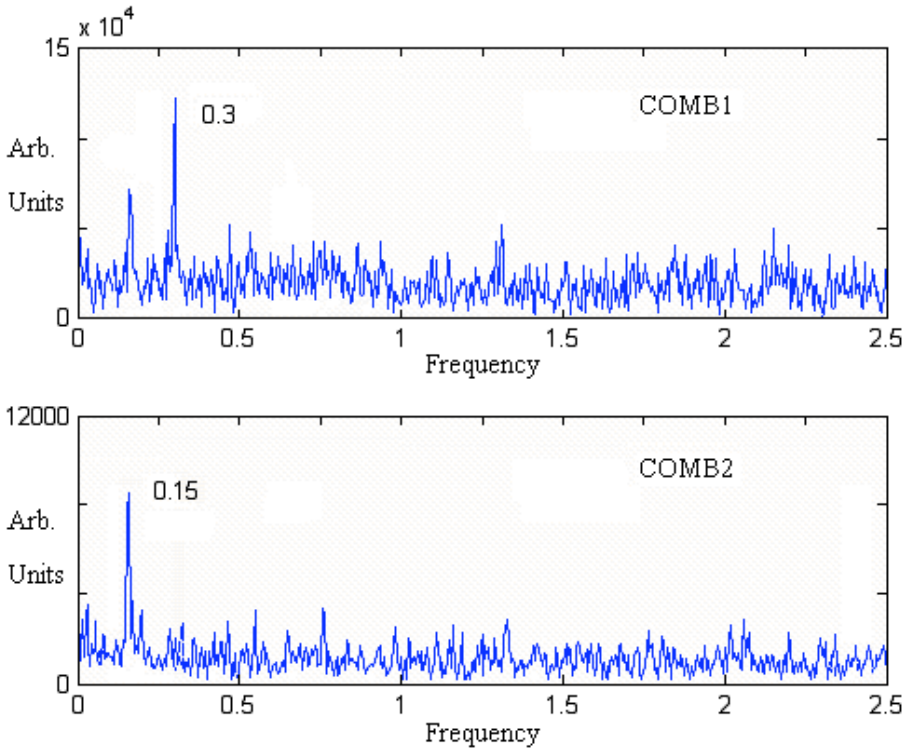

Fig.4.1.3 Simulation at an Approximate Human Breathing Rate

4.2 Summary and Conclusions

In this work, the author seeks to demonstrate a combined STM/HOME sensor based on previous research for the application of the 'smart' bed to non-intrusively monitor a patient's movement and vital signs. The emphasis is on the development of hardware and the modification of software for this specific kind of spatially distributed integrating fiber optic sensor. A number of tasks have been fulfilled and are summarized as follows:

(1) First, a literature review was done on the history of fiber optical sensors. Due to their low cost, lightweight, small size and other attractive properties, fiber optical sensors have been extensively developed over 30 years and have become widely used in various research and industrial fields. A review of fundamental knowledge on optical waveguides was also presented. The basics of electromagnetic fields that are solutions to Maxwell's equations are of great importance in understanding, modeling and evaluating electromagnetic energy transmission in optical fibers.

(2) Electromagnetic wave theory and slab waveguides have been used for many years to explain the properties of light transmission in fiber. In the second chapter, a theoretical analysis was given for multi-mode wave propagation in optical waveguides. Different point of view based on both wave theory and geometric optics were presented on mode conversion with respect to external perturbation upon the fiber.

(3) Mathematical modeling of the STM and HOME sensors approaches was analyzed and the advantage of a combined STM/HOME sensor was discussed. The STM sensor represents a modal interference approach, while the HOME signal is processed via that of mode conversion. With their combination, it was shown that the dynamic range of spatially distributed multi-mode optical sensors can be extended.

(4) An optical element was introduced to generate uniform high order modes, an essential hardware development for the realization of the HOME modal modulation approach.

(5) A modification was made to the data processing software since the combined STM/HOME sensor contains dual processing outputs. The software allows simultaneous real-time output of both the STM and HOME signals.

(6) Finally, experiments simulating human body breathing and heartbeat rates by periodic mechanical perturbations were carried out with the use of the combined STM/HOME sensor. Of the two modal modulation approaches, the COMB1 output was more sensitive to small perturbations such as heartbeats but the signal saturates for large perturbations such as major patient movements. Comparatively, the COMB2 signal has less distortion and shows more advantage under large perturbations.

In future work, we will miniaturize the high order mode generation bulk optics through the use of a configuration of GRIN lenses inside the signal-conditioning unit. We also plan to explore the application of this sensor technology in hospital intensive care units to determine its practicality in monitoring the motion of unconscious patients.

References:

[1] X. Xu, W.B. Spillman Jr., R.O. Claus, K.E. Meissner, and K. Chen, "Spatially distributed fiber sensor with dual processed outputs", to be presented at the 17th International Conference on Optical Fiber Sensors, Bruge (May, 2005).

[2] X. Xu, K.E. Meissner and W.B. Spillman Jr., "Modification of fiber optic sensors for 'smart' bed", presented at the 2004 Optics in the Southeast Conference, Raleigh-Durham (November, 2004).

[3] W.B. Spillman et al, "A 'smart' bed for non-intrusive monitoring of patient physiological factors", J. Meas. Sci. Tech. nol. 15, 2004.

[4] A. Melloni, F. Carniel, "Determination of Bend Mode Characteristics in Dielectric Waveguides", Journal of Lightwave Technology, Vol. 19, No. 4, 2001.

[5] W.B. Spillman Jr. and D.R. Huston, "Scaling and antenna gain in integrating fiber optic sensors", Journal of Lightwave Technology, Vol.13, No. 7, pp.1222-1230, 1995.

[6] P.R. Herczfeld et al, "An embedded fiber optic sensor utilizing the modal power distribution technique", J. Opt. Lett. 15, No. 21, pp.1242-1244, 1990.

[7] W.B. Spillman et al, "Statistical mode sensor for fiber optic vibration sensing applications", Applied Optics 28, No. 15, pp.3166-3176, 1989.

[8] John A. Buck, Fundamentals of Optical Fibers, 2nd edition, 2004.

[9] Robert J. Hoss, Edward A. Lacy, Fiber Optics, 2nd edition, 1993.

[10] Udd, Fiber Optic Sensors—An Introduction for Engineers and Scientists, 1991

[11] Alen H. Cherin, An Introduction to Optical Fibers, 1983

[12] Luc B. Jeunhomme, Single-Mode Fiber Optics—Principles and Applications, 1983.

[13] Takanori Okoshi, Optical Fibers, 1982.

[14] N. S. Kapany, Fiber Optics; Principles and Applications, 1967.

Appedix

```
// SpeckleDlg.cpp : implementation file
//

#include "stdafx.h"
#include <math.h>
#include "Speckle.h"
#include "SpeckleDlg.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

#define Show_LB_x 0
#define Show_LB_y 200
#define Show_Width 400
#define Show_Height 180
#define MaxISample 10000
LRESULT CALLBACK FrameCallbackProc(HWND hWnd, LPVIDEOHDR lpVHdr);
BYTE bFrame[2][230404],pixel[240][320],mpixel[240][320];
BOOL bAutoScale,bAutoScale2;
static int iFrame, iArray, SliderShow, SliderShow2, threshold;
static int ix_c,iy_c,radius,cright,cleft,ctop,cbottom;
static long SPD[MaxISample],iarea_c[MaxISample];
static long inum,ix_sum,iy_sum,itotal,maxc,maxSPD,MaxShow,MaxShowc;
CWnd *pWnd, *pSPDWnd, *pWndSlider,*pWndSlider2;
CDC *pDC, MemDC;
CBitmap MemBitmap;
CSliderCtrl m_Slider,m_Slider2;
CBitmap *pOldBit;
HWND m_hCapWnd;// Preview window
/////////////////////////////////////////////////////////////////////////////
// CSpeckleDlg dialog

CSpeckleDlg::CSpeckleDlg(CWnd* pParent /*=NULL*/)
	: CDialog(CSpeckleDlg::IDD, pParent)
	, m_iFrame(0)
	, m_SPD(0)
	, m_iFrame2(0)
	, m_Centre(0)
{
	//{{AFX_DATA_INIT(CSpeckleDlg)
	//}}AFX_DATA_INIT
	// Note that LoadIcon does not require a subsequent DestroyIcon in
Win32
	m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
}

void CSpeckleDlg::DoDataExchange(CDataExchange* pDX)
```

```
{
	CDialog::DoDataExchange(pDX);
	//{{AFX_DATA_MAP(CSpeckleDlg)
	//}}AFX_DATA_MAP
	DDX_Text(pDX, IDC_FRAME1, m_iFrame);
	DDX_Text(pDX, IDC_SPDValue, m_SPD);
	DDX_Text(pDX, IDC_FRAME2, m_iFrame2);
	DDX_Text(pDX, IDC_CENTRE, m_Centre);
}

BEGIN_MESSAGE_MAP(CSpeckleDlg, CDialog)
	//{{AFX_MSG_MAP(CSpeckleDlg)
	ON_WM_PAINT()
	ON_WM_QUERYDRAGICON()
	ON_WM_TIMER()
	ON_BN_CLICKED(IDC_SOURCE, OnSource)
	ON_BN_CLICKED(IDC_FORMAT, OnFormat)
	ON_BN_CLICKED(IDC_SAVE, OnSave)
	ON_WM_VSCROLL()
	ON_BN_CLICKED(IDC_CHECK1, OnCheck1)
	ON_BN_CLICKED(IDC_CHECK2, OnCheck2)
	//}}AFX_MSG_MAP

END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
// CSpeckleDlg message handlers

BOOL CSpeckleDlg::OnInitDialog()
{
	CDialog::OnInitDialog();

	// Set the icon for this dialog.  The framework does this automatically
	//  when the application's main window is not a dialog
	SetIcon(m_hIcon, TRUE);			// Set big icon
	SetIcon(m_hIcon, FALSE);		// Set small icon

	// TODO: Add extra initialization here
	m_bInit=FALSE;
	MaxShow=10000000;
	UpdateData(false);
	bAutoScale=FALSE;
	bAutoScale2=FALSE;
	pWnd=AfxGetMainWnd()->GetDlgItem(IDC_VIDEO);
	pSPDWnd=AfxGetMainWnd()->GetDlgItem(IDC_SPD);
	pDC=pSPDWnd->GetDC();
	MemDC.CreateCompatibleDC(NULL);
		//Create a bitmap compatible with screen display
	MemBitmap.CreateCompatibleBitmap(pDC,Show_Width+50,Show_Height*2+50);
	pOldBit=MemDC.SelectObject(&MemBitmap);
	SetTimer(ID_RecordingTimer,100, NULL);
	iFrame=-1;
	iArray=0;
	maxSPD=100;
	maxc=100;
```

```
	inum=0;
	ix_c=0;
	iy_c=0;
	cright=0;
	cleft=0;
	ctop=0;
	cbottom=0;
	m_pRedPen   = new CPen(PS_SOLID, 1, 0x000000ff);
	m_pBlackPen   = new CPen(PS_SOLID, 1, 0x00010101);
	m_pBkClrPen   = new CPen(PS_SOLID, 1, 0x00d0d0d0);
       CRect rect; // Get pointer to preview window
	pWnd->GetWindowRect(&rect); // Get size of the window
	m_hCapWnd=capCreateCaptureWindow((LPTSTR)TEXT("Wireless Sensor"),
		WS_CHILD|WS_VISIBLE|WS_EX_CLIENTEDGE|
		WS_EX_DLGMODALFRAME,0,0,rect.Width(),rect.Width(),
		pWnd->GetSafeHwnd(),0);
	//Setup display window
	ASSERT(m_hCapWnd);
	if(capDriverConnect(m_hCapWnd,0)){ //Connect device 0
		m_bInit=TRUE;
		// Get properties of device
		capDriverGetCaps(m_hCapWnd,sizeof (CAPDRIVERCAPS), &m_CapDrvCap);
		if(m_CapDrvCap.fCaptureInitialized){// If initialization succeeds
			capGetStatus(m_hCapWnd, &m_CapStatus,sizeof
(m_CapStatus)); // get state of driver
			capPreviewRate(m_hCapWnd,30); // Set sample period with
unit mS
			capPreview(m_hCapWnd,TRUE); // set preview style
		}
		else{// if initialization fails
			AfxMessageBox("Can't find video source!");
			AfxGetMainWnd()->PostMessage(WM_CLOSE);
		}
	}
	else{// Can't connect to driver
		AfxMessageBox("Can't find video source!");
		AfxGetMainWnd()->PostMessage(WM_CLOSE);
	}
	capSetCallbackOnFrame(m_hCapWnd,FrameCallbackProc);

	pWndSlider =GetDlgItem( IDC_SLIDER); //CRect rect;
	pWndSlider->GetWindowRect( &rect );
	ScreenToClient( &rect );
	m_Slider.Create(
WS_VISIBLE|WS_CHILD|TBS_VERT|TBS_RIGHT|TBS_BOTH|TBS_AUTOTICKS,
                                rect, this, IDC_SLIDER );
	m_Slider.SetTicFreq( 10 );    // Send TBM_SETTICFREQ
	m_Slider.SetLineSize( 1 );       // Send TBM_SETLINESIZE
	m_Slider.SetPageSize( 10 );       // Send TBM_SETPAGESIZE
	m_Slider.SetRange( 0, 100, TRUE ); // Send TBM_SETRANGE
	//////////////////// Initialise two controls
	pWndSlider2 =GetDlgItem( IDC_SLIDER2);
	pWndSlider2->GetWindowRect( &rect );
	ScreenToClient( &rect );
```

```
	m_Slider2.Create(
WS_VISIBLE|WS_CHILD|TBS_VERT|TBS_RIGHT|TBS_BOTH|TBS_AUTOTICKS,
                                    rect, this, IDC_SLIDER2 );
	m_Slider2.SetTicFreq( 10 );
	m_Slider2.SetLineSize( 1 );
	m_Slider2.SetPageSize( 10 );
	m_Slider2.SetRange( 0, 100, TRUE );
	return TRUE;  // return TRUE  unless you set the focus to a control
}

// If you add a minimize button to your dialog, you will need the code
// below to draw the icon.  For MFC applications using the document
// /view model, this is automatically done for you by the framework.

void CSpeckleDlg::OnPaint()
{	int i;
	if (IsIconic())
	{
		CPaintDC dc(this); // device context for painting

		SendMessage(WM_ICONERASEBKGND, (WPARAM) dc.GetSafeHdc(), 0);
		// Center icon in client rectangle
		int cxIcon = GetSystemMetrics(SM_CXICON);
		int cyIcon = GetSystemMetrics(SM_CYICON);
		CRect rect;
		GetClientRect(&rect);
		int x = (rect.Width() - cxIcon + 1) / 2;
		int y = (rect.Height() - cyIcon + 1) / 2;
		// Draw the icon
		dc.DrawIcon(x, y, m_hIcon);
	}
	else
	{	CPaintDC dc(this); // device context for painting
		m_iFrame=iFrame;
		m_iFrame2=iFrame;
		m_SPD=SPD[iFrame];
		m_Centre=iarea_c[iFrame];
		UpdateData(false);
		pDC=pSPDWnd->GetDC();

		//Fill area with background color
	MemDC.FillSolidRect(0,0,Show_Width,Show_Height*2+41,RGB(236,233,216));
	     //Draw coordinate
		MemDC.SelectObject(m_pBlackPen);
		MemDC.MoveTo(Show_LB_x,Show_LB_y);
		MemDC.LineTo(Show_LB_x,Show_LB_y-Show_Height);
		MemDC.MoveTo(Show_LB_x,Show_LB_y);
		MemDC.LineTo(Show_LB_x+Show_Width,Show_LB_y);

		MemDC.MoveTo(Show_LB_x,Show_LB_y*2);
		MemDC.LineTo(Show_LB_x,Show_LB_y*2-Show_Height);
		MemDC.MoveTo(Show_LB_x,Show_LB_y*2);
		MemDC.LineTo(Show_LB_x+Show_Width,Show_LB_y*2);
	     //To erase the former graph
		MemDC.SelectObject(m_pBkClrPen);
```

```
		if (iFrame<Show_Width) {
			MemDC.MoveTo(Show_LB_x, Show_LB_y-
int(SPD[0]*1.0/maxSPD*Show_Height));
			for(i=1;i<=iFrame-1;i++) {
				MemDC.LineTo(i+Show_LB_x, Show_LB_y-
int(SPD[i]*1.0/maxSPD*Show_Height));
			}
		}
		else {
			MemDC.MoveTo(Show_LB_x, Show_LB_y-int(SPD[iFrame-
Show_Width-1]*1.0/maxSPD*Show_Height));
			for(i=1;i<=Show_Width-1;i++) {
				MemDC.LineTo(i+Show_LB_x, Show_LB_y-int(SPD[iFrame-
Show_Width+i-1]*1.0/maxSPD*Show_Height));
			}
		}  ////////////////////////////////////////For COMB1

		if (iFrame<Show_Width) {
			MemDC.MoveTo(Show_LB_x, Show_LB_y*2-
int(iarea_c[0]*0.8/maxc*Show_Height));
			for(i=1;i<=iFrame-1;i++) {
				MemDC.LineTo(i+Show_LB_x, Show_LB_y*2-
int(iarea_c[i]*0.8/maxc*Show_Height));
			}
		}
		else {
			MemDC.MoveTo(Show_LB_x, Show_LB_y*2-int(iarea_c[iFrame-
Show_Width-1]*0.8/maxc*Show_Height));
			for(i=1;i<=Show_Width-1;i++) {
				MemDC.LineTo(i+Show_LB_x, Show_LB_y*2-
int(iarea_c[iFrame-Show_Width+i-1]*0.8/maxc*Show_Height));
			}
		}  ////////////////////////////////////For COMB2

		MemDC.SelectObject(m_pRedPen);   //To draw the graph
		if (iFrame<Show_Width) {
			if (bAutoScale) {
				maxSPD=100;
				for(i=1;i<=iFrame;i++) maxSPD = maxSPD>SPD[i]?
maxSPD: SPD[i];
				UpdateData(false);
			}
			else  maxSPD=MaxShow;

			MemDC.MoveTo(Show_LB_x, int(Show_LB_y-
SPD[0]*1.0/maxSPD*Show_Height));
			for(i=1;i<=iFrame;i++) {
				MemDC.LineTo(i+Show_LB_x, Show_LB_y-
int(SPD[i]*1.0/maxSPD*Show_Height));
			}
		}
		else {
			if (bAutoScale) {
				maxSPD=100;
```

```
                        for(i=iFrame-Show_Width;i<=iFrame-1;i++) maxSPD =
maxSPD>SPD[i]? maxSPD: SPD[i];
                        UpdateData(false);
                    }
                    else maxSPD=MaxShow;
                    MemDC.MoveTo(Show_LB_x, Show_LB_y-int(SPD[iFrame-
Show_Width]*1.0/maxSPD*Show_Height));
                    for(i=1;i<=Show_Width-1;i++) {
                        MemDC.LineTo(i+Show_LB_x, Show_LB_y-int(SPD[iFrame-
Show_Width+i]*1.0/maxSPD*Show_Height));
                    }
            }  ////////////////////////////////////For COMB1

            if (iFrame<Show_Width) {
                    if (bAutoScale2) {
                        maxc=100;
                        for(i=1;i<=iFrame;i++) maxc = maxc>iarea_c[i]?
maxc: iarea_c[i];
                        UpdateData(false);
                    }
                    else  maxc=MaxShowc;

                    MemDC.MoveTo(Show_LB_x, int(Show_LB_y*2-
iarea_c[0]*0.8/maxc*Show_Height));
                    for(i=1;i<=iFrame;i++) {
                        MemDC.LineTo(i+Show_LB_x, Show_LB_y*2-
int(iarea_c[i]*0.8/maxc*Show_Height));
                    }
            }
            else {
                    if (bAutoScale2) {
                        maxc=100;
                        for(i=iFrame-Show_Width;i<=iFrame-1;i++) maxc =
maxc>iarea_c[i]? maxc: iarea_c[i];
                        UpdateData(false);
                    }
                    else maxc=MaxShowc;
                    MemDC.MoveTo(Show_LB_x, Show_LB_y*2-int(iarea_c[iFrame-
Show_Width]*0.8/maxc*Show_Height));
                    for(i=1;i<=Show_Width-1;i++) {
                        MemDC.LineTo(i+Show_LB_x, Show_LB_y*2-
int(iarea_c[iFrame-Show_Width+i]*0.8/maxc*Show_Height));
                    }
            }  ////////////////////////////////////For COMB2

            pDC->BitBlt(0,0,Show_Width,Show_Height*2+41,&MemDC,0,0,SRCCOPY);
            //clear after draw
            CDialog::OnPaint();
        }
}

// The system calls this to obtain the cursor to display while the
// user drags the minimized window.
HCURSOR CSpeckleDlg::OnQueryDragIcon()
{
```

```
       return (HCURSOR) m_hIcon;
}

LRESULT CALLBACK FrameCallbackProc(HWND hWnd, LPVIDEOHDR lpVHdr)
{      long i,j,k;
      long double l;
       iFrame++;
       if (iFrame>=MaxISample) {
              for(i=0;i<=Show_Width-1;i++) {
                     SPD[i]=SPD[iFrame-Show_Width+i];
                     iarea_c[i]=iarea_c[iFrame-Show_Width+i];
              }
              iFrame=Show_Width;
       }
       memcpy(bFrame[iArray],lpVHdr->lpData,(lpVHdr->dwBufferLength)/1);
//Copy data from the buffer to bFrame[iArray]
       inum=(lpVHdr->dwBufferLength)/1;
       for(i=0;i<240;i++) {
              for(j=0;j<320;j++) {
            pixel[i][j]=bFrame[iArray][i*320+j];
                     mpixel[i][j]=bFrame[iArray][i*320+j];
              }
       }
       threshold=180;
    for(i=0;i<240;i++) {
              for(j=0;j<320;j++) {
            if (mpixel[i][j]<threshold) mpixel[i][j]=0;
              }
       }
       ix_sum=0;
       iy_sum=0;
       itotal=0;
       if (iFrame<31) {
              for (i=0;i<240;i++){
                     for(j=0;j<320;j++){
                  ix_sum+=j*pixel[i][j];
                     iy_sum+=i*pixel[i][j];
                     itotal+=pixel[i][j];
                     }
              }
           ix_c=int(ix_sum/itotal);
           iy_c=int(iy_sum/itotal);
        // To find the center of intensity by the equation of "weighted
       // mean value" of all the simplified data of the intensity
       // distribution.

       for (i=iy_c;mpixel[i][ix_c]==0,i>0;i--) ctop=i;
       for (i=iy_c;mpixel[i][ix_c]==0,i<240;i++) cbottom=i;
       for (i=ix_c;mpixel[iy_c][i]==0,i>0;i--) cleft=i;
       for (i=ix_c;mpixel[iy_c][i]==0,i<360;i++) cright=i;
       iy_c=int((ctop+cbottom)/2); //find the middle of vertical lines
       ix_c=int((cleft+cright)/2);//find the middle of horizontal lines
       radius=int(min(cright-ix_c,cbottom-iy_c));
       }
```

```
        iarea_c[iFrame]=0;
        if(ix_c>radius && iy_c>radius && ix_c<320-radius && iy_c<240-radius){
               for (i=iy_c-radius;i<iy_c+radius+1;i++) {
                      l=radius^2-(iy_c-i)^2;
                      k=int(sqrt(l));
                  for (j=ix_c-k;j<ix_c+k+1;j++){
                         iarea_c[iFrame]+=pixel[i][j];
                      }
               }
        }
        else    iarea_c[iFrame]=0;
        // total intensity of central area

        SPD[iFrame]=0;
        for (i=0;i<76800;i++) {
               SPD[iFrame]+=abs(bFrame[0][i]-bFrame[1][i]);
        }      //  intensity change of entire frame

     iArray=1-iArray;
        return (LRESULT) TRUE;
}

void CSpeckleDlg::OnOK()
{
        // TODO: Add extra validation here
        MemBitmap.DeleteObject();
        MemDC.DeleteDC();
        capDriverDisconnect(m_hCapWnd);
        CDialog::OnOK();
}

void CSpeckleDlg::OnTimer(UINT nIDEvent)
{
        // TODO: Add your message handler code here and/or call default
        SendMessage (WM_PAINT,0,0 );

        CDialog::OnTimer(nIDEvent);
}

void CSpeckleDlg::OnSource()
{
        // TODO: Add your control notification handler code here
        capDlgVideoSource(m_hCapWnd);
}

void CSpeckleDlg::OnFormat()
{
        // TODO: Add your control notification handler code here
        capDlgVideoFormat(m_hCapWnd);
}

void CSpeckleDlg::OnSave()
{      OPENFILENAME ofn;
        FILE * stream;
        long i;
```

```
        char szFileName[MAX_PATH]= "";
        ZeroMemory(&ofn, sizeof(ofn));
        ofn.lStructSize = sizeof(ofn);
        ofn.hwndOwner = m_hCapWnd;
       ofn.lpstrFilter = "Text Files (*.txt)\0*.txt\0All Files (*.*)\0*.*\0";
        ofn.Flags = OFN_EXPLORER | OFN_PATHMUSTEXIST | OFN_HIDEREADONLY |
OFN_OVERWRITEPROMPT;
        ofn.nMaxFile = MAX_PATH;
       ofn.lpstrFile = szFileName;
        ofn.lpstrDefExt = "txt";
        if(GetSaveFileName(&ofn)) {
               stream = fopen( szFileName, "w" );
               fprintf(stream, "center_x: %ld\t\tcenter_y: %ld\t\tradius:
%ld\n\n", ix_c, iy_c, radius);
               fprintf( stream, "STM:\t\t\t\tHOME:\n\n");
               for (i=0;i<=iFrame;i++)
               fprintf( stream, "%ld\t\t\t\t%ld\n", SPD[i],iarea_c[i]);
               fclose( stream );
    }
}

void CSpeckleDlg::OnVScroll(UINT nSBCode, UINT nPos, CScrollBar* pScrollBar)
{
        // TODO: Add your message handler code here and/or call default
        SliderShow=m_Slider.GetPos();
        MaxShow=(100-SliderShow)*100000;
        SliderShow2=m_Slider2.GetPos();
        MaxShowc=(100-SliderShow2)*100000;
        UpdateData(false);
        CDialog::OnVScroll(nSBCode, nPos, pScrollBar);
}

void CSpeckleDlg::OnCheck1()
{
        // TODO: Add your control notification handler code here
        bAutoScale=~bAutoScale;
//      bScaleChanged=TRUE;
}

void CSpeckleDlg::OnCheck2()
{
        // TODO: Add your control notification handler code here
        bAutoScale2=~bAutoScale2;
}
```

Vita

The author, Xiaohua Xu was born in January 1981 in Shanghai, People's Republic of China. She received her Bachelor of Science degree in 2003 from Fudan University, Shanghai, majored in Physics. In August 2003 the author came to the Physics Department of Virginia Polytechnic Institute and State University for her graduate study and worked as a teaching assistant for the physics department. During January 2004 to August 2004, she was doing her internship in Virginia Tech Applied Bioscience Center, Blacksburg, Virginia. The subject of the research in the internship was taken as the research project for her Master of Science degree in Physics Department.