Reconstructing Specimens using DIC Microscope Images

Farhana Kagalwala   Takeo Kanade

Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
farhana@cs.cmu.edu   tk@cs.cmu.edu

Abstract

Differential Interference Contrast (DIC) microscopy is a powerful visualization tool used to study live biological cells. Its use, however, has been limited to qualitative observations. The inherent nonlinear relationship between the object properties and the image intensity makes quantitative analysis difficult. Towards quantitatively measuring optical properties of objects from DIC images, we develop a method to reconstruct the specimen's optical properties over a three-dimensional volume. The method is a nonlinear optimization which uses hierarchical representations of the specimen and data. As a necessary tool, we have developed and validated a computational model for the DIC image formation process. We test our algorithm by reconstructing the optical properties of known specimens.

1 Introduction

The Nomarski differential interference contrast (DIC) microscope is the preferred method for visualizing live biological specimens. The DIC microscope is an interferometer, and therefore the refractive structure of the specimen is made visible. In biological research, live, transparent cells can be imaged with this microscope modality. Three-dimensional structure can be visualized by optically-sectioning[1] through the specimen. To date, however, biologists only qualitatively assess DIC images of cell specimens. Quantitative microscopy methods, such as computational optical sectioning microscopy (COSM), have been restricted to linear microscopy modalities.[5] The inherent nonlinearities in the DIC image formation process have hindered past attempts at quantitative analysis. In this paper, we describe a method to reconstruct specimens imaged with DIC microscopy.

DIC microscopy offers several advantages over other contrast-generating optical systems. In DIC the pupil is unobstructed, and therefore transverse[2] and axial resolution exceeds that in Zernike phase contrast. Consequently, thick specimens with three-dimensional

---

[1] For each image in an optically sectioned set, the optical elements are configured to focus at a particular object distance.

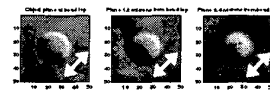[2] Transverse planes are perpendicular to the optical axis of the microscope

Figure 1: *DIC Image of Bead: Specimen has a $10\mu m$ polystyrene bead embedded in optical cement. The leftmost image is in focus while the other 2 images are increasingly de-focused. The arrow indicates the shear.*

features are better resolved. Unlike some fluorescence methods, no dyes are injected and therefore live specimens are not adversely affected. Finally, unlike confocal scanning methods which have slow rates of acquisition, an entire stack of optically sectioned images can be acquired within a minute. In the case of mobile cells, the short acquisition time minimizes distortions between optically sectioned slices.

Looking through the eyepiece of the DIC microscope, an observer sees a shadow cast image which deceptively indicates three-dimensional structure. Example DIC images are shown in Fig. 1. Actually, the image is the differential of the optical path length introduced by the object into the propagating light wave. The differential is along a particular direction, in the transverse plane, called the shear direction. In addition, each image contains both in-focus and out of focus information. Therefore, the challenge in DIC microscopy remains to interpret image features belonging to in-focus object properties correctly.

2 DIC Microscopy Background

The DIC microscope, is essentially a brightfield microscope with a polarizer-analyzer pair and two prisms.[Fig. 2] As in standard brightfield optics, light from a lamp is collimated by a collector and a condenser lens combination. In DIC, however, a polarizer and a Wollaston prism is inserted between the collector and condenser lens. Moreover, the prism is positioned with respect to the back focal plane of the condenser. The described setup produces two mutually coherent, polarized beams. Each electric field is polarized perpendicularly with respect to the other. In
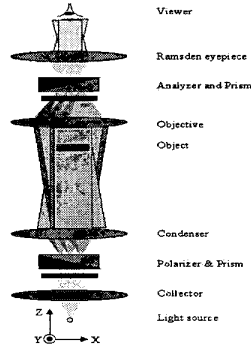
Figure 2: *DIC Optical Components: Regular bright-field microscope components such as a light source, collector, condenser, objective and eyepiece are supplemented with a pair of polarizer-prism set. Three light paths are shown to illustrate conjugate planes of reference. Optical elements, spacing, and the incident light angles are not to scale.*

addition, the wavefronts impinging on the object are differentially translated with respect to each other. In front of the objective lens, an analyzer and Nomarski prism are inserted and aligned with the front focal plane of the objective. The Wollaston prism behind the condenser introduces a linear phase gradient across the two fields emerging from the condenser. The Nomarski prism in front of the objective lens compensates for this linear phase gradient. Therefore, the combined action of the two prisms results in a constant phase bias between the two perpendicularly polarized fields. This combination produces a steady pattern of interference between the two beams which can then be detected by a CCD camera or the human eye.

Mathematically, the DIC imaging process can be summarized by the following set of equations. First, consider a coherent field represented by

$$U_{ill}(\vec{x}) = A \exp[-i\phi_{ill}(\vec{x})]. \qquad (1)$$

where $\vec{x} = (x, y, z)$ are spatial variables in the coordinate space of the object with the z-axis coincident with the optical axis. $\phi(\vec{x})$ is the phase function. The action of the two prisms can be represented as an aggregate by $U_{ill}^1(\vec{x})$ and $U_{ill}^2(\vec{x})$, where

$$
\begin{aligned}
U_{ill}^1(\vec{x}) &= A \exp\left[-i[\phi_{ill}(\vec{x} + \vec{v}_s) + \Delta\phi_{bias}]\right] & (2) \\
U_{ill}^2(\vec{x}) &= A \exp\left[-i\phi_{ill}(\vec{x} - \vec{v}_s)\right] & (3)
\end{aligned}
$$

In the above equations, $\Delta\phi_{bias}$ is the constant phase bias and $2\vec{v}_s$ is the shear vector. After a phase transformation, $\phi_{obj}(\vec{x})$, due to the object, the wavefronts

$$
\begin{aligned}
U_{obj}^1(\vec{x}) &= A \exp\left[-i[\phi_{obj}(\vec{x} + \vec{v}_s) + \Delta\phi_{bias}]\right] & (4) \\
U_{obj}^2(\vec{x}) &= A \exp\left[-i\phi_{obj}(\vec{x} - \vec{v}_s)\right]. & (5)
\end{aligned}
$$

contain the object information. The field in the image space is

$$
U_{img}(\vec{x}_{img}) = \int_{y_0^{obj}}^{y_1^{obj}} \int_{x_0^{obj}}^{x_1^{obj}} K_z(x_{img}, y_{img}; x, y)
$$
$$
[U_{obj}^1(x, y, z = z_1) - U_{obj}^2(x, y, z = z_1)] dx dy
$$

where $\vec{x}_{img} = (x_{img}, y_{img}, z_{img})$ are spatial variables in the coordinate space of the image with the z_{img}-axis coincident with the optical axis. $K_z(x_{img}, y_{img}, x, y)$ is the complex amplitude point spread function of the imaging system (objective and any other auxiliary lens) of the microscope. $K_z(...)$ describes the propagation from the object plane at $z = z_1$ to the image plane z_{img}. The image intensity $I(\vec{x}_{img})$ is

$$I(\vec{x}_{img}) = |U_{img}(x_{img}, y_{img}, z_{img})|^2 \qquad (6)$$

The nonlinearity in the DIC image has two basic sources. First, since the image is an interference pattern, the detected intensity is the squared magnitude of the light field's complex amplitude. Therefore, a convolution of the light intensity with a lens transfer function does not accurately represent the DIC image, which is a linear superposition of complex amplitude, rather than intensity components, of the light field. In addition, out of focus contributions from the object have to be considered. Therefore a three-dimensional amplitude point spread (or transmission) function is needed to accurately model the image intensity. Second, the object itself aberrates the light wave as it propagates through. The biological specimens under consideration, though weakly refractive, are thick and scatter light significantly. Therefore, aberrations due to the object contribute significantly to the image.

Deconvolution methods, such as computational optical sectioning microscopy (COSM), are widely used to recover object information from images acquired by certain optical modalities. COSM methods model the image intensity as a convolution of the object's intensity transmittance with a computed point-spread function. In modalities such as fluorescence and brightfield, a linear function of the intensity from the object provides an accurate, first order approximation of the image. However, in DIC microscopy, the image cannot be represented by merely considering the intensity from the object. Both phase and amplitude information have to be modelled.

Initial work in DIC image analysis algorithms used linear models.[4] In 1996, Feinegle used a contour finding algorithm to locate edges in each image from an optically sectioned stack.[6] In her work, specimen structure is obtained by axial stacking of contours. The most recent work in the analysis of DIC images is by Preza.[12] This work recovers the optical pathlength at each image point due to the object, and therefore does recover some quantitative information. The only other attempt to quantify information from DIC optics has been made by Cogswell, et. al. using optical techniques referred to as geometric phase

methods.[3] Though we are unaware of cases where this method has been applied to recover three dimensional object information. The work by Feinegle produced a three dimensional model of the specimen, but the object properties were not quantitated and the recovered specimen model was not validated with ground truth experiments. Preza's work recovers optical path-length but does not actually reconstruct three dimensional object information. So far, no attempt has been made to quantitatively reconstruct the three dimensional properties of the object from DIC images.

In contrast, the reconstruction algorithm that we have developed recovers the whole object information. We address the nonlinearities in the image formation process by a precise computational model of DIC microscopy which is used for reconstructing specimens. Using a generalized ray tracing method we have developed a model of light propagation through the microscope and specimen. [8, 9] The computational model is used to generate simulated images of the current estimated object as part of the hierarchical reconstruction algorithm.

3 Reconstruction Problem Description

Given a set of DIC images, the goal of the reconstruction is to estimate the refractive index distribution throughout the volume encompassing the object. Due to the nonlinearity of the DIC image-formation process, a direct inversion of the imaging equations is not feasible. Therefore, we use an iterative nonlinear optimization algorithm. The optimization starts with an initial estimate of the distribution. Using our computational model, we generate simulated images corresponding to this estimate. The sum of squared differences between the real data and the simulated images is the error function. The estimate is modified at each iteration such that this error function is minimized. Such a reconstruction process requires a representation of the refractive index distribution at every point across the specimen volume. One possible representation is a weighted combination of basis functions where the weights are the parameters. The degrees of freedom in the representation, that is the number of parameters used, plays a critical role in the accuracy with which an object can be represented and the ease with which the optimization converges to the correct answer. If too few parameters are used, the object may not be represented with enough accuracy. If too many parameters are used, the optimization algorithm could be overwhelmed and may not converge to the solution.

A nonlinear optimization method traverses the parameter space searching for the point where the error function is minimized. To avoid converging to incorrect minima, traditional nonlinear optimization problems require that the initial set of parameters be close to the actual solution. In general, the number of local minima increases with more degrees of freedom in the representation, since the error function develops more undulations. Finding an appropriate initial estimate for complex optimization problems is difficult. In order to be able to converge to the correct solution, despite the initial estimate being far, one needs a sys-

tematic method of traversing the parameter space such that the estimates approach the neighborhood of the global minima without being trapped in local minima. One possible method to achieve this is to first reduce the parameters such that only coarse object properties can be represented. If the volumetric distribution is approximated by a small number of parameters, the consequent error function is also of reduced dimension. In the process, local fluctuations (minima) are smoothed over and the error function retains global shape properties. One can then solve the optimization with respect to the reduced parameter set. When projected back to the original problem, the solution of the coarse optimization is closer to the original global minima. Thus the coarse solution serves as an appropriate initial estimate for the original optimization problem. Depending on the complexity of the problem, this parameter reduction can be done in multiple stages.

In our method, we represent the object and image data hierarchically and optimize at successive levels of the hierarchy by using a multi-level wavelet-like representation for the object. At each level, the original image data is projected onto a basis spanning a space of possible image features at the given object resolution level. This projected data is compared with simulated images in our nonlinear optimization. Thus, at each level of the hierarchical reconstruction we obtain an object estimate that fits the data at that level. This object is used as an initial estimate for the optimization at the next level of the hierarchy.

4 Reconstruction Algorithm

The hierarchical reconstruction consists of three processes. In the first step, we represent the object at different resolutions. Second, we decompose the image data in order to identify image features corresponding to objects at the different resolutions. Third is the nonlinear optimization algorithm which recovers an estimate of the object at each resolution.

4.1 Hierarchical Object Representation

Since the DIC image captures the directional gradient of the phase introduced by the object into the light path, we represent the object as a volumetric distribution of refractive index values. At present, we assume that the object is transparent, and therefore the refractive index is a scalar quantity.[†] We represent the refractive index by a wavelet-like bases, approximating it at different levels of resolution, where at each level only a subset of spatial frequencies are present. For the purpose of illustration, let us formulate in one dimension. At a particular resolution J, $n_J(x)$ is represented by combining versions of the scaling function, $\Phi(x)$, and wavelet function, $\Psi(x)$,

$$n_J(x) = \Sigma_{k=-2}^{K-2} a_k \Phi(x-k) + \Sigma_{j=0}^{J} \Sigma_{k=-1}^{2^j K-2} b_{j,k} \Psi(2^j x-k) \tag{7}$$

Both $\Phi(x)$ and $\Psi(x)$ are cubic spline functions, developed by Cai and Wang[2]. The scaling function

---

[†] This may be generalized to include absorption. The refractive index would then be a vector, or complex number.

represents the lowest spatial frequencies. The wavelet function represents the details at different resolutions. Translated versions of $\Phi(x)$, in addition to, scaled and translated versions of $\Psi(x)$ span the domain of an object. To represent a function at a particular level one needs to find the combination of weights up to that level which best approximates the original function. In Fig. 3, at the first level (J=-1), an example function is coarsely approximated by a combination of the translated scaling functions. At each subsequent level, the corresponding wavelet functions are added to the representation and the function is better approximated. To represent a two-dimensional distribution, $n_J(x, z)$, we have to use a combination of functions that are outer products of the one-dimensional scaling and wavelet functions,

$$
\begin{aligned}
n_J(x, z) \;=\; & \sum_{k=-2}^{K-2} \sum_{l=-2}^{L-2} a_{k,l}\Phi(x - k)\Phi(z - l) \\
& + \sum_{j=0}^{J} [\sum_{k=-1}^{2^jK-2} \sum_{l=-2}^{2^jL-2} b_{j,k,l}\Psi(2^j x - k)\Phi(2^j z - l) \\
& + \sum_{k=-2}^{2^jK-2} \sum_{l=-1}^{2^jL-2} c_{j,k,l}\Phi(2^j x - k)\Psi(2^j z - l) \\
& + \sum_{k=-1}^{2^jK-2} \sum_{l=-1}^{2^jL-2} d_{j,k,l}\Psi(2^j x - k)\Psi(2^j z - l)](8)
\end{aligned}
$$

Each tensor product results in a two-dimensional function that can represent different properties of the object. [Fig. 4] The extension to three dimensions is similar. The levels of the wavelet functions provide a systematic hierarchy for representing and recovering the object. At each level, of the reconstruction algorithm, the goal is to recover the parameters at that level. The scaling function coefficients $(a_{k,l})$ are recovered in the very first level (J=-1). At that level, the distribution is

$$
n_{-1}(x, z) = \sum_{k=-2}^{K-2} \sum_{l=-2}^{L-2} a_{k,l}\Phi(x - k)\Phi(z - l) \qquad (9)
$$

The values of $a_{k,l}$ are initialized to 0. We then proceed to update the parameters $a_{k,l}$ iteratively so that the error between the simulated images and the image data at this level is minimized. The iterations at this level terminate when small changes in the parameter vector do not effect a significant improvement to the error. The algorithm then proceeds to the next level (J=0). The estimated object from the previous level is transferred as the initial object at this level and we proceed to recover $b_{0,k,l}$, $c_{0,k,l}$, $d_{0,k,l}$. Analogous to the previous level, these parameters are initialized to 0 and iteratively updated so that the error between simulated images and the image data at level 0 is minimized. Each subsequent level of the reconstruction proceeds similarly.
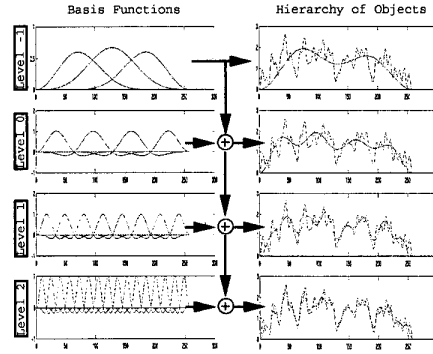

Figure 3: *Object Hierarchy. In the right column an original 1-d object is shown superimposed with the approximate object at the level of the hierarchy. The left column shows the basis functions at each level. Each row shows the basis functions that are added to the representation at that level.*
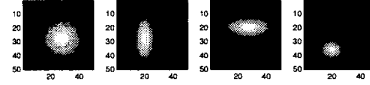

Figure 4: *Sample 2-d Scaling and Wavelet Functions. Each corresponds to a term in Eq.8.*

The wavelet-like basis provides a continuous representation for the refractive index. For computations, we discretize this distribution. In our model, the refractive index is stored at discrete locations of a three-dimensional voxel grid. We typically use grids with resolution .2-1 μm perpendicular to the optical axis and .03-1 μm axially. A typical grid size is 100x100x50 in the x, y and z dimension respectively.

4.2 Hierarchical Image Representation

Unlike microscope modalities for which analytical models of the three-dimensional optical transfer function are available[13], object structure and the DIC image is related nonlinearly. Due to this nonlinearity, an algorithm using an analytical DIC imaging model is not practical. If the relationship between the object and image intensity were linear, then a linear decomposition of the object would result in a linear decomposition of the image. Since this is not possible, we have developed a "matching by synthesis" algorithm that identifies image features which result from the represented object frequencies. The original data consists of image features due to all object frequencies but this data cannot be linearly decomposed into features at all levels. Therefore, at each resolution level we have to determine the image features which are appropriate and match these to the original data. An
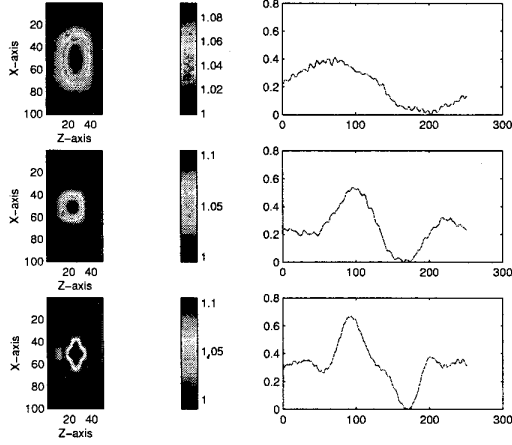
Figure 5: *Hierarchy of Objects and Images. The 2 object dimensions are the axial (Z) and transverse (X). For each image, the object plane is at the center of the Z axis. Rows 1-3 show the objects and corresponding images from the coarsest to finest resolution (J=-1, J=0, J=1).*

example of an object at different resolutions and the corresponding images simulated by our computational model is depicted in Fig. 5.

The matching by synthesis algorithm extracts features in the original intensity data which correspond to objects at a particular resolution. At a given resolution level J, we suppose that the DIC image of the object can be represented in terms of a basis such that

$$I_J(x_{img}, z_{img}) = \sum_{q=0}^{Q^J} r_q^J B_q^J(x_{img}, z_{img}) \qquad (10)$$

where $\{B_q^J\}$ for $q = 0..Q^J$ is a set of basis functions specific to the resolution J. In order to relate the original image data, $I(x_{img}, z_{img})$, to the images at each resolution level, we need to find the features in the original data that correspond to the features in the basis functions. Before we can proceed to match features in the original data, we need to find the basis images. We obtain the basis images by perturbing parameters at a particular resolution and synthesizing an image per perturbed object. Functions selected to be in the basis represent the image features most common to the synthesized images. The original image data is then projected onto this basis, thereby giving the image features that are present in the original data corresponding to an object at the given resolution. Specifically, our algorithm proceeds in three steps. First, at a particular resolution, we introduce a large set of perturbations to the initial object and generate simulated images of each of the perturbed objects. Second, we select a basis that best characterizes the images corre-

sponding to the perturbed objects. Third, we project the original image data onto the estimated basis.

4.2.1 Object Perturbations

At a given resolution level J, the initial object is perturbed by a large number of random configurations. Each perturbation involves adding or subtracting a random amount from all parameters of the Jth level basis functions. Therefore, given random parameter vectors b^{rand}, c^{rand} and d^{rand}, a perturbed object at level J is

$$n_J^{pert}(x, z) = n_{J-1}(x, z)$$

$$+ \sum_{k=-2}^{2^J K - 2} \sum_{l=-1}^{2^J L - 2} (b_{J,k,l} + b_{k,l}^{rand}) \Psi(2^J x - k) \Phi(2^J z - l)$$

$$+ \sum_{k=-1}^{2^J K - 2} \sum_{l=-2}^{2^J L - 2} (c_{J,k,l} + c_{k,l}^{rand}) \Phi(2^J x - k) \Psi(2^J z - l)$$

$$+ \sum_{k=-2}^{2^J K - 2} \sum_{l=-2}^{2^J L - 2} (d_{J,k,l} + d_{k,l}^{rand}) \Psi(2^J x - k) \Psi(2^J z - l) \quad (11)$$

We use the computational model to generate a simulated image, $I_{sim,J}(x_{img}, z_{img})$, for each perturbed object. The set of simulated images contain a wide range of possible image features corresponding to objects at this resolution. Though, we have only shown perturbations of the wavelet coefficients, the coefficients of the scaling functions are also perturbed at level J=-1 to obtain basis images at that level. For a particular resolution, the total degrees of freedom at that resolution is

$$\alpha_J = (2^J K)(2^J L - 1) + (2^J K - 1)(2^J L) + (2^J K)(2^J L). \tag{12}$$

We have experimentally determined that $8\alpha_J$ perturbations (and simulated images $I_{sim,J}(x_{img}, z_{img})$) suffice.

4.2.2 Basis Selection

Next, we implement a basis selection method from the simulated images, $\{I_{sim,J}^i(x_{img}, z_{img})\}$ for $i = 1..8\alpha_J$. One simple basis selection method is a Karhunen-Louve decomposition applied to two-dimensional signals.[10, 14] This method results in a set of "eigenimages", each functioning as a basis vector. We found that the eigenimages based method does not sufficiently abstract image features from the set of images. That is, although each $I_{sim,J}(x_{img}, z_{img})$ can be represented by a combination of the eigenimages, each eigenimage doesn't necessarily represent isolated image features.

To find basis vectors which explicitly represent image features, we implemented a basis selection method using the "matching pursuit" algorithm. [11, 1, 7] This algorithm uses a redundant dictionary of functions, where each function is a scaled and translated version

of an exponentially modulated window function. In our implementation, the window is Gaussian. Using a redundant dictionary of such functions offers an advantage over decomposition into a pre-established basis, such as a wavelet basis. In a wavelet basis, scale and frequency have a fixed relationship so that only features which have a particular frequency content can be represented at a particular scale. In contrast, exponentially modulated functions can represent several kinds of features at any given scale.

The matching pursuit is an iterative greedy algorithm to identify functions out of the dictionary that best match an image in the set. Note, the iterations of the matching pursuit algorithm are embedded within each resolution level of the reconstruction algorithm. Given an intensity vector $I(x_{img})$, $(I(x_{img}) = I^i_{sim,J}(x_{img}, z_{img} = constant)$, $i \in [1, 8\alpha_J]$ in our algorithm), the matching pursuit algorithm defines a residual, $R_w\{I\}(x_{img})$ at each iteration w, where $R_0\{I\}(x_{img}) = I(x_{img})$. An iteration consists of finding, the best function $g_b^w(x_{img})$ from candidate functions in the dictionary, such that

$$< R_w\{I\}(x_{img}), g_b^w(x_{img}) >> < R_w\{I\}(x_{img}), g_i(x_{img}) > \tag{13}$$

where $<>$ denotes inner product. $g_b^w(x_{img})$ is then removed from the candidate functions in the dictionary. The residual is updated as,

$$R_{w+1}\{I\}(x_{img}) = R_w\{I\}(x_{img})$$
$$- < R_w\{I\}(x_{img}), g_b^w(x_{img}) > g_b^w(x_{img}). \tag{14}$$

The algorithm is terminated when the residual's norm falls below a preset threshold. More details can be found in the references.

As a result of matching pursuit, each $I^i_{sim,J}$ is decomposed with respect to a unique matched set, $\{g_b^w(x_{img})\}$ for $w = 1..w_{max}$, selected from the dictionary. Therefore we will have $8\alpha_J$ matched sets. Finally, we extract a set of dissimilar functions that are most prevalent across all the matched sets.[1] This set of functions forms the basis, $\{B_q^J\}$ for $q = 0...Q^J$, for the subspace of images corresponding to objects at resolution level J. [Fig. 6] The number, Q^J, is set experimentally to ensure that $\{B^J\}$ spans the entire image.

4.2.3 Image Projection

Once the appropriate basis hierarchy for the images is selected, the original image data is projected onto the basis at each level. A matrix, M_J, is constructed in which all the columns are the individual functions comprising of the basis at a particular resolution.

---

[1] Two functions, $g_1(x_{img})$ and $g_2(x_{img})$, are dissimilar if $\frac{<g_1,g_2>}{<g_1,g_1>}$ and $\frac{<g_1,g_2>}{<g_2,g_2>}$ is less than an empirically determined threshold.
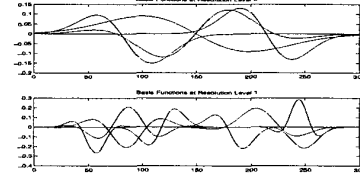
Figure 6: *1-d basis functions at the levels 0 and 1. (From specimen 1 reconstruction).*

That is,

$$M_J = \begin{bmatrix} B_0^J(x, z = 0) & ... & B_{Q^J}^J(x, z = 0) \\ B_0^J(x, z = 1) & ... & B_{Q^J}^J(x, z = 1) \\ \vdots & & \\ B_0^J(x, z = z_{max}) & ... & B_{Q^J}^J(x, z = z_{max}) \end{bmatrix} \tag{15}$$

In the above matrix, each basis image corresponding to a different axial position is stacked vertically so that the entire basis image can be represented as a column vector. Since the matrix will not be square in most cases, the original data is multiplied with the pseudo-inverse to obtain the necessary weights for each of the basis functions. This is the over constrained solution of $\frac{d}{d\vec{r}}\{M_J\vec{r} - I\}^2 = 0$ where $\vec{r}$ is the vector of weights, I is the original data, formatted into a column vector similar to the basis images above. The least squares solution is $\vec{r} = M_J^\dagger I$. where $M_J^\dagger$ is the Moore-Penrose pseudo-inverse of M_J. The projected image, I_J,

$$I_J = P_J\{I\} = M_J\vec{r} \tag{16}$$

corresponding to the Jth resolution is the weighted linear combination of the basis functions, using the functions obtained with the matching pursuit algorithm and the weights based on the least squares solution.

4.3 Optimization at Each Resolution

For each level in the resolution hierarchy, we optimize with respect to the wavelet coefficients at the current resolution. At a particular level, the initial object is the estimated object from the previous level. At the first resolution level, the initial object is a completely blank volume, i.e. all voxels are initialized with the same refractive index as the background. The image data used in the optimization is the original data projected onto the basis selected for this resolution level. (I_J defined previously.) So at each resolution level, the target data is actually the projected images described above. At the final resolution level, we use the original data. The sum of squared differences between the projected data and the simulated data is the error term for the optimization.

4.3.1 Levenberg-Marquardt

At the first resolution level, J=-1, the degrees of freedom of the object is quite low. Therefore,

a Levenberg-Marquardt type nonlinear optimization produces sufficiently good results. [15] This is basically a Gauss-Newton type gradient-based optimization, where the Hessian matrix is supplemented with an identity matrix scaled by a parameter. The parameter is decreased at each iteration if the current estimate reduces the error, otherwise it is increased. Implicitly, this optimization method assumes that the neighborhood of the global solution is predominantly convex, barring some minor local undulations, and that the current estimate is within this neighborhood.

4.3.2 Randomized Methods

At subsequent resolution levels, the degrees of freedom in the object increase rapidly. No longer can one assume that the initial estimate is close to the solution at this level and that the neighborhood along the error surface to be traversed is sufficiently convex. Therefore, Gauss-Newton type methods such as Levenberg-Marquardt tend to converge to local minima. Although, the optimization is significantly aided by the resolution hierarchy, we still need to implement a randomized search method within the nonlinear optimization. We have implemented a genetic algorithm that has been very successful in finding the correct solutions.

The genetic algorithm combines parameters from members of a population to produce a new member. The population that we use to initiate the genetic algorithm is the same set of perturbed parameter vectors described above. Each parameter vector, p, is a concatenation of $b_{J,k,l}$, $c_{J,k,l}$ and $d_{J,k,l}$. In addition, each p has an associated error value which is the sum of squared differences between the simulated image due to this object and the projected image at the current level, plus any penalty terms due to constraints. The population is divided into good and bad candidates, represented by p_{good}^k and p_{bad}^k. The good and bad candidates have error values in the top and bottom 50th percentile respectively. There are two ways of combining member parameters. The first method selects two good members (p_{good}^1 and p_{good}^2) and a bad candidate (p_{bad}^1). The new member is $p_{new}^{comb} = a(.5p_{good}^1 + .5p_{good}^2) - bp_{bad}^1$ where $a > 0$, $b > 0$ and $a - b = 1$. This method is similar to traditional combination methods in genetic algorithms. In this selection method, good candidates with lower error values have better chances of being selected than ones with higher error values. If the new member is better than either of its "parents",i.e. p_{good}^1 and p_{good}^2, according to the above defined error, then the worst parent is replaced by the new member. The second method simply selects a member of the population at random and randomly perturbs some parameter values, i.e. $p_{new}^{mutate} = p^k + p_{rand}$ where p^k is a randomly selected parameter vector from the entire population and p_{rand} is a vector of random values. This method is consistent with traditional mutation. If the error value corresponding to p_{new}^{mutate} is better than that of the worst parameter vector, then this mutant replaces
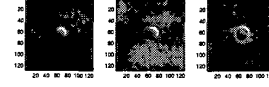

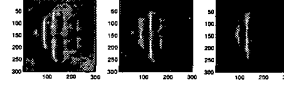Figure 7: *Physical data of bead specimen.*

Figure 8: *Real data of wafer specimen.*

the worst member. At each iteration of the algorithm we sort the members according to their error values, and perform one combination and one mutation step.

4.3.3 Constraints

The only strong constraint on the object is that the minimum refractive index be 1.0. Therefore, every estimated distribution is offset by a suitable amount to ensure that the minimum value in the distribution is 1.0. In our reconstruction experiments, the physical specimen has only two distinct refractive index values. So for those experiments we add a term in the error that penalizes for inhomogeneity in the distribution. This additional penalty term is only introduced at the finest resolution level and has considerably sped up convergence. The penalty is achieved by segmenting the volume into regions of high and low refractive index values and then measuring the variance within those regions. The variance terms are scaled by parameters, ω and β and added to the error. Therefore the modified error is

$$
\begin{aligned}
E = & \sum_{m=1}^{M} (I_J[m] - I_{sim}[m])^2 \\
& + \frac{\omega}{N_{high}} \sum_{=\gamma_{high}} (n[k] - n_{high})^2 \\
& + \frac{\beta}{N_{low}} \sum_{k=\gamma_{low}} (n[k] - n_{low})^2 \quad (17)
\end{aligned}
$$

where the first term is the sum of squared differences between the intensity values of the projected image and the simulated image calculated over the total number of pixels, M. In the rest of the error, N is the total number of voxels, n_{high} and n_{low} are the means of the high and low valued n[x] respectively, γ_{high} and γ_{low} are the sets of discrete voxels which are above and below the threshold respectively.

5 Results

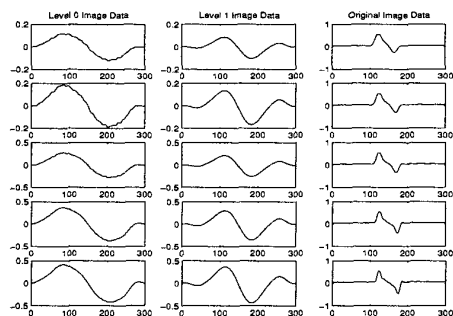The reconstruction results we have obtained thus far have all been in two dimensions. That is, in-

Figure 9: *Intensity plots at different resolutions. Column 1-2: The original data projected onto the levels 0-1 respectively. Column 3: The original data.*
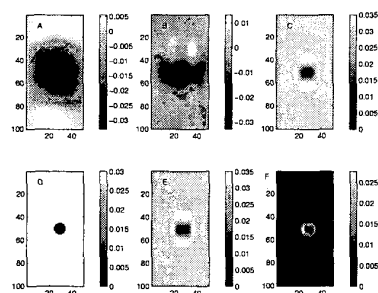

Figure 10: *Results for Specimen 1. A-C are the estimated answers at levels 0-2. D: Real object. E: Final estimate. F: Error between real and estimated object.*
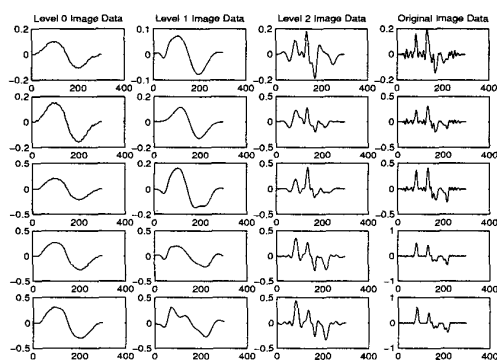

Figure 11: *Intensity plots of wafer data at different resolutions. Columns 1-3: The projected data at levels 0-2 respectively. Column 4: The original data.*
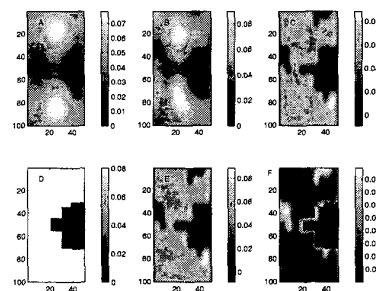

Figure 12: *Results for Specimen 2. A-C are the estimated answers at levels 0-2. D: Real object. E: Final estimate. F: Error between real and estimated object.*

stead of estimating the full three-dimensional volumetric distribution, we chose to reconstruct one slice in the volume. To provide a good proof of concept for this reconstruction algorithm that would be less computationally expensive than a full three-dimensional reconstruction, we chose an axial slice (rather than a transverse slice). Our decision is based on the fact that reconstructing along the axial dimension presents the greatest challenge. Spatial frequencies which have significant components in the axial direction are attenuated the most in the image formation process. The two transverse dimensions are imaged according to identical principles, thus reconstructing in one of them can serve as a validation for both. We chose two test specimens, each having some unique attributes. The first specimen, consisting of a bead in optical cement, is symmetric and the object structure is embedded in the volume. The second specimen, consisting of an ion-milled wafer, does not have perfect symmetry and the structure extends to the top edge of the volume.

In the first experiment, we embedded several 10μm diameter beads in optical cement. The homogeneous beads have a refractive index that is .03 less than the cement. We acquired an optically sectioned data set of this specimen with an axial resolution of .2 μm. A cropped region showing one bead in some of the images is shown in Fig. 7. The data used in the reconstruction experiment consisted of a diagonal cut through the cropped region of 10 images in the set. Some of the intensity plots are shown in Fig. 9. The image data projected onto the basis at different object resolutions is shown in Fig. 9 as well. Fig. 10 shows the estimated objects at the different resolutions and a comparison between the expected object structure and the recovered object. The grid resolution in this experiment is 1μm in both directions. As can be seen from the error image, the object structure is recovered quite accurately. The original object has symmetric structure and the recovered shape is symmetric as well. The specimen used for the second experiment consisted of an ion-milled glass wafer. Due to error prone multiple millings, the actual milled wafer pattern is defective

in that it does not have symmetric walls even though the specifications have symmetric structure. The specimen was prepared by filling the milled cavity with oil that had a refractive index .08 less than the glass. The optically sectioned image set of this specimen has a .2μm axial resolution. A subset of the images are shown in Fig. 8. For the reconstruction experiment, we extracted a line of intensity data from each one of ten images in the set. The line (horizontal) was extracted above the midpoint of the shown images at a point such that two of the depressions in the pattern are captured. The original image data is shown in Fig. 11 along with the results of the image data projected onto the basis at different object resolutions. Fig. 12 shows the estimated object at different resolutions and a comparison of the final reconstructed object and the expected object. The grid resolution in this experiment is .3 in the x-direction and .03 in the z-direction. Since the exact nature of the defects introduced by the milling process is not known, the expected object shown is merely a hypothesis. As can be seen, barring some small extraneous patches, the structure of the recovered object is very close to that of the expected object. The lack of symmetry that is apparent from the DIC images appears in the recovered object as well.

6 Discussion and Conclusion

The analysis of DIC images presents a substantial challenge due to the nonlinearity of the image formation process and the out-of-focus artifacts. We tackle the problem by developing two tools. First, the computational model has been discussed in detail in previous publications. The second is the hierarchical reconstruction algorithm discussed here. As shown by the results, the nonlinear optimization is powerful enough to recover axial and transverse structure and quantitate the optical properties of the object. Even though, we initialize the optimization far from the actual solution, we are able to converge very close to it. In order to successfully converge to solutions when faced with highly nonlinear error surfaces, we developed a hierarchical method. This method represents the object with respect to a wavelet basis in order to systematically reduce the dimension of the search space and arrive at a chain of object estimates at finer and finer resolutions. The intensity data also has to be decomposed in a manner consistent with the object decomposition. Since the imaging process is nonlinear, we developed a method by which image features present at the different object resolutions are explicitly captured in a particular basis at each level. The real data can then be projected onto the basis at each resolution level to obtain a hierarchical representation of the data. Our results show that such an algorithm is capable of recovering structure along all directions. By recovering the structure of two very different specimens, we illustrate the capabilities of the reconstruction algorithm. We are currently performing reconstruction experiments in all three dimensions. Since the computational model is already capable of simulating three dimensional objects, the extension merely applies to the hierarchical reconstruction method.

References

[1] F. Bergeaud and S. Mallat. Matching pursuit:adaptive representations of images and sounds. *Comp. Appl. Math.*, 15(9):97–109, 1996.

[2] W. Cai and J. Wang. Adaptive multiresolution collocation methods for initial boundary value problems of nonlinear pdes. *SIAM J. of Num. Anal.*, 33(3):937–970, 1996.

[3] C.J. Cogswell, N. Smith, K. Larkin, and P. Hariharan. Quantitative dic microscopy using geometrical phase shifter. *Proc. of SPIE*, 2984:72–81, 1997.

[4] Kristen Dana. Three dimensional reconstruction of the tectorial membrane. Master's thesis, MIT, 1992.

[5] A. Erhardt and et. al. Reconstructing 3-d light microscopic images by digital image processing. *App. Op.*, 24(2):194–200, 1985.

[6] Patricia Feinegle. *Motion analysis and visualization of biological structures imaged via Nomarski DIC Light microscopy*. PhD thesis, CMU, 1996.

[7] S. Jaggi and et. al. High resolution pursuit for feature extraction. *App. and Comp. Harm. Anal.*, 5:428–449, 1998.

[8] F. Kagalwala, F. Lanni, and T. Kanade. Simulating DIC microscope images. *Proc. of IEEE Wkshp. on Photo. Modelling in Comp. Vision and Graph.*, June, 1999.

[9] F. Kagalwala, F. Lanni, and T. Kanade. Computational model of DIC microscopy: From observations to measurements. *CMU-RI TR*, TR-00-10, 2000.

[10] M. Kirby and L. Sirovich. Applications of the karhunen-louve procedure for the characterization of human faces. *IEEE PAMI*, 12:103–108, 1990.

[11] S. Mallat and Z. Zhang. Matching pursuits with timefrequency dictionaries. *IEEE Trans. on Sig. Proc.*, 41(12):3397–3410, 1993.

[12] Chrysanthe Preza. *Phase Estimation Using Rotational Diversity for DIC Microscopy*. PhD thesis, Wash. U., 1998.

[13] N. Streibl. Three-dimensional imaging by a microscope. *JOSA A*, 2(2):121–127, 1985.

[14] M. Turk and A. Pentland. Eigenfaces for recognition. *J. of Cog. Neuro.*, 3(1):71–86, 1991.

[15] W.Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, 2 edition, 1992.