Errata

- Page 31, line 12 should refer to 'section §2.7' and not 'appendix §2.7'
- Section §2.7 in the dissertation should be corrected to refer to the following:

2.7 Strain Tracking for GAIM

Accurate estimation of full-field displacements from motion capture of soft tissue deformation is critical to the successful application of GAIM as an analysis technique. As the tissue deforms continuously over multiple stretch protocols, it is required that the displacement fields be available for the continuous deformation. Errors in displacement field estimation have detrimental effects on the inversion of material parameters– the inverse matrix is a function of strains, and any noise in displacements is amplified in the strains due to the gradient operator. Furthermore, our method of inversion is direct, making it more sensitive to errors in displacement fields than to iterative methods that use repeated solves to match displacements. In this chapter we briefly outline our method for accurate estimation of displacement and strain fields for a tissue sample undergoing multiple biaxial stretch protocols. We make use of a popular image registration technique– digital image correlation, and combine it with optical flow, finite element discretization, and some heuristic algorithms.

2.7.1 Digital Image Correlation (DIC)

DIC is an image registration technique that determines the location of matching image subsets drawn from two or more images. Given two image subsets, the pixel array intensities of the two are correlated against each other with a correlation function whose maxima corresponds to the location of the best match between the two

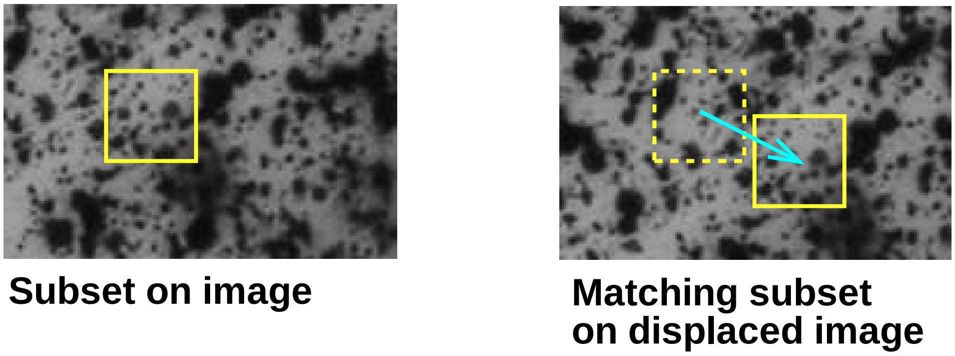


Figure 2.1: Basic principle of digital image correlation: A subset from one image is matched in a second image. The displacement vector between the centers of subsets indicates the net displacement that the first subset undergoes.

subsets. Figure 2.1 illustrates the matching subsets in two images and the effective displacement vector that corresponds to the rigid body translation between the two images. A detailed review of various DIC techniques for estimating displacements and strains is provided in Refs. [78,79] where the authors compared the performance of correlation algorithms in spatial versus Fourier domains (e.g., [80,81,82]).

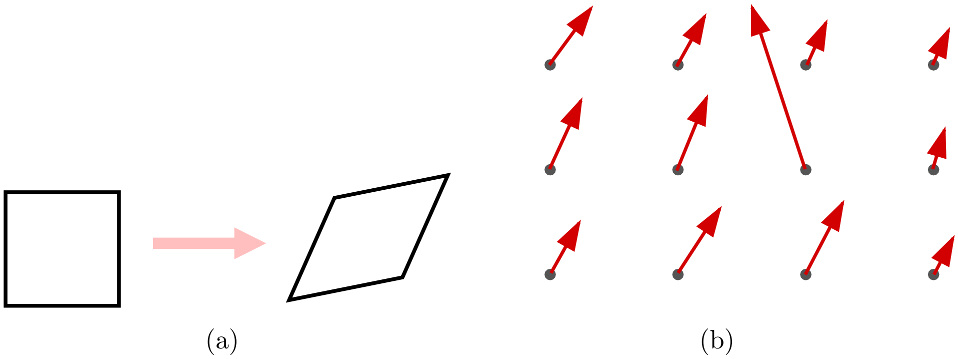


Figure 2.2: Issues that simple DIC methods cannot handle: (a) Deformation of the subset or ROI is common in mechanical tests of soft tissues, (b) Poor visual texture or inadequate match of subsets due to ROI deformation may lead to points that are badly tracked.

While most basic DIC methods handle rigid body translations adequately, they are less accurate when the subset (also known as region of interest, ROI) undergoes deformation as shown in Fig. 2.2(a). Mismatch due to not accounting for deformation of the ROI may result in poor tracking such as the one shown in Fig. 2.2(b). Various subpixel algorithms are reviewed in [83,79] where it was shown that ROI deformation can be accounted for by an optical flow method, and the subpixel displacement calculated with high accuracy by a Newton-Raphson optimization of the correlation criterion. Pan *et al.* [84] provide an approximation to the optimization method that achieves sufficiently high accuracy without the need for expensive Hessian computations. We adopt the iterative least-squares method detailed in [84] and briefly summarize the key equations here.

Given two subsets f and g, where the former is of size (2M+1)×(2M+1) and the latter is at least the same size, the zero-normalized cross correlation (ZNCC) function between the two is computed as

$$ZNCC(u,v) = \sum_{i=-M}^{M}\sum_{j=-M}^{M} \frac{(f(x_i,y_j)-f_m)(g(x_i-u,y_j-v)-g_m)}{\Delta f \Delta g} \tag{2.1}$$

$$\Delta f = \sqrt{\sum_{i=-M}^{M}\sum_{j=-M}^{M}(f(x_i,y_j)-f_m)^2} \tag{2.2}$$

$$\Delta g = \sqrt{\sum_{i=-M}^{M}\sum_{j=-M}^{M}(g(x_i-u,y_j-v)-g_m)^2} \tag{2.3}$$

where f_m and g_m are the mean values of the respective subsets, u and v are the shift from the center of the template g. If the two templates are perfectly matched then $ZNCC$ would have a large peak at the center corresponding to $(u,v)=(0,0)$ indication no shift of template. We use the implementation of ZNCC with the MATLAB® function `normxcorr2`.

Assuming that the mapping between the ROI in f and g is accompanied by a linear shift and scaling of the light intensity, then

$$af(x_i,y_i)+b=g(x_i',y_i') \tag{2.4}$$

where a and b are the linear scaling and shift factors of light intensity spectrum, and (x_i',y_i') are the corresponding locations in g of point (x_i,y_i) in f. We assume that a homogeneous strain state is present in the subset g, and by using a first-order Taylor series approximation we have,

$$x_i' = x_i + u + \Delta u + u_x \Delta x_i + u_y \Delta y_i \tag{2.5}$$

$$y_i' = y_i + v + \Delta v + v_x \Delta x_i + v_y \Delta y_i \tag{2.6}$$

where u and v are the integer pixel components of displacement, Δu and Δv are the subpixel components, and the remaining terms are associated with the first-order gradients. These equations can be cast into a linear least-squares system with the vector of unknown parameters, p, and the mimization function, F given as

$$p = \begin{bmatrix} \Delta u & u_x & u_y & \Delta v & v_x & v_y & a & b \end{bmatrix}^T \tag{2.7}$$

$$F_i(p) = g\left(x_i'(p), y_i'(p)\right) - af\left(x_i(p), y_i(p)\right) - b \approx 0 \tag{2.8}$$

The system in Eqn. 2.8 can be solved iteratively for unknown p with the least-squares approach as

$$p^{k+1} = p^k - \left[\nabla F(p^k)^T \nabla F(p^k)\right]^{-1} \nabla F(p^k)^T F(p^k) \tag{2.9}$$

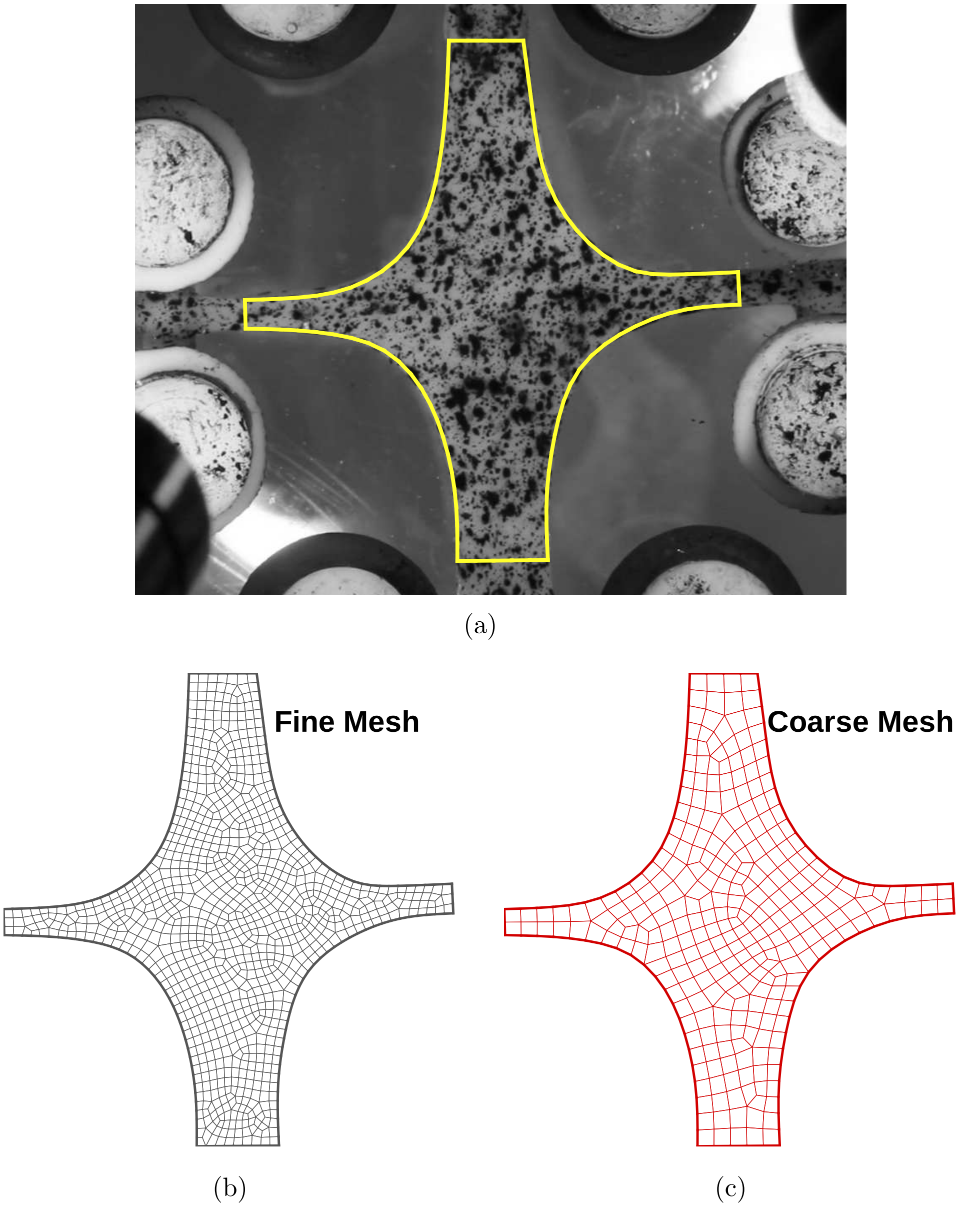
Hence, the integer pixel displacement of a pixel location on image f can be determined by matching a subset of appropriate size drawn from f and centered at the location of interest, with a larger subset that encompasses the possible location in image g. The location of the peak of the ZNCC function between the two, roughly corresponds to the location of the matching pixel and the displacement is vector computed from the net movement.

2.7.2 Mesh Based Guidance for Image Correlation

In order to avoid badly tracked points such as the one shown in Fig. 2.2(b), we make use of information from tracked neighbors to guess the location where a candidate point might have moved. The total displacement of a pixel can be split into three components as

$$U_{total} = U_{ng} + U_{zncc} + U_{sp} \tag{2.10}$$

where U_{ng} is the displacement guessed from tracked neighbors, U_{zncc} is the displacement from ZNCC for a subset centered at the guessed location in g, and U_{sp} is the subpixel displacement computed by Eqns. 2.4–2.9 for a subset in g, centered at the location shifted from the guess by the result from ZNCC.



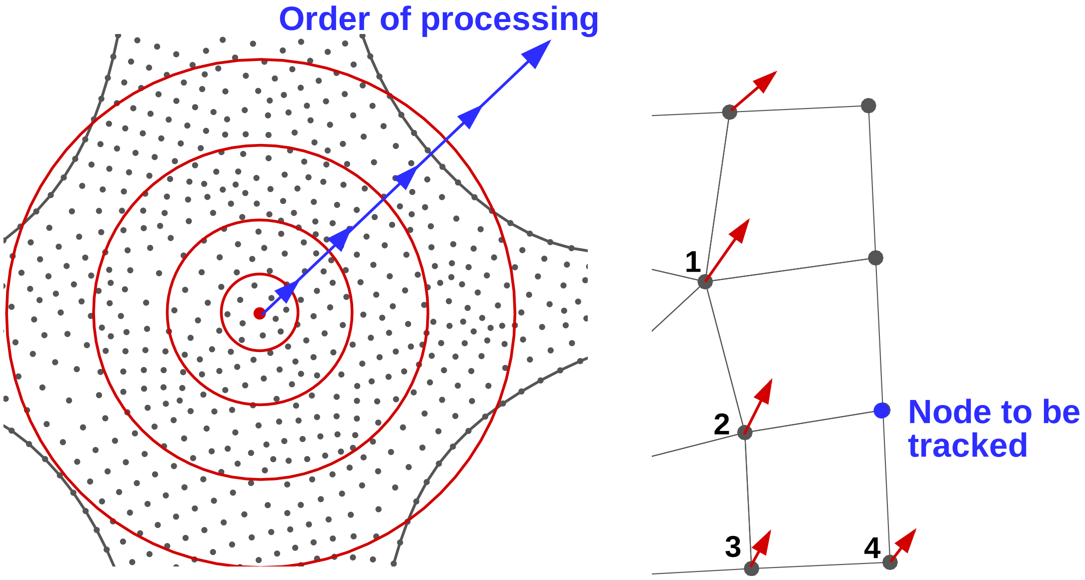
(a)

(b)

(c)

Figure 2.3: (a) The boundary of the tissue domain of interest is sketched on top of the reference image, and two meshes are generated. (b) Nodes of the fine-mesh are tracked by correlation to supply a dense estimation of the displacement field. (c) Displacement fields are interpolated onto the coarse-mesh which is used for the GAIM analysis.

Because GAIM requires displacements on a finite-element domain, we use a mesh that is mapped on top of the image corresponding to the reference state of the tissue (Fig. 2.3(a)). The boundary of the tissue based on the reference image is sketched in ABAQUS® and two quadrilateral meshes are created (Figs. 2.3(b),2.3(c)). The fine-mesh is used to perform the image correlation and to obtain the displacement field. The field is interpolated onto the coarse-mesh which is used for the GAIM analysis. Use of a mesh provides several benefits, namely: (a) in contrast to using a grid of locations, the mesh can be chosen to match the shape of the tissue domain which is often irregular, (b) rapid determination of tracked neighbor nodes is possible by storing mesh adjacencies, and (c) strains computed from displacements on the irregular domain using finite-element theory are more accurate than finite-difference gradient approximations on a grid. The displacements of tracked neighbor nodes are averaged and the value used as a guess for the location of the candidate node in g.



(a)

(b)

Figure 2.4: Mesh based guidance for image correlation: (a) Mesh nodes are processed in the order of increasing distance from a user-specified guide-point (red node). Red concentric circles indicate the zones of increasing distance. (b) Candidate node to be tracked is marked in blue; tracked nodes in its vicinity have red displacement vectors. The numbered nodes are neighbors to the blue node and share an element with it. The average displacement vector of the numbered nodes is used to estimate the location of the blue node.

Figure 2.4(a) illustrates the order of processing mesh nodes. The user provides an initial input of the region with best texture or least movement, and is asked to confirm or reject the approximate location of movement that is estimated solely by ZNCC. Upon acceptance, a list of candidate nodes is made in the order of increasing distance from this guide-node. Each candidate node in the list is processed in sequence so that it has tracked nodes in its neighborhood. To illustrate, the red node in Fig. 2.4(a) is the user-specified guide point, the red concentric circles indicate the regions in increasing distance from this point and correspond to the order of processing. Figure 2.4(b) shows the candidate node marked in blue and surrounded by tracked nodes in its neighborhood (marked with displacement vectors). The adjacent tracked nodes which share an element with the candidate node are numbered. The displacement vectors of the numbered nodes are averaged to obtain a guess, U_{ng}, for the movement of the candidate node. It is straight-forward to incorporate an error-checking mechanism to avoid bad tracked points such as the one shown in Fig. 2.2(b). All our tissues of interest are planar continua, therefore we can set a maximum cut-off for U_{zncc} which would prevent large deviation of displacements from its surrounding. We perform this operation in two steps:

1. During the first round of correlation, any node whose U_{zncc} exceeds a cut-off is treated as poorly tracked and marked for later examination. Its displacement is not used to provide guesses for neighbors.
2. When all nodes in the list have been processed, the poorly tracked nodes are returned to. The average displacement of tracked neighbors is recomputed and treated as the sum of $U_{ng} + U_{zncc}$. The reasoning behind this approach is that if there are patches with poor texture or dirt, the patch is avoided, its surrounding is processed instead, and the patch is filled at the end when tracked neighbors surround it completely.

The subset sizes for U_{zncc} and U_{sp} are not necessarily the same. We denote the subset sizes for ZNCC and subpixel by M_{zncc} and M_{sp} respectively. It is obvious that M_{zncc} should be large enough to contain distinctive texture that can be matched, and small enough to keep computation times inexpensive. Likewise, M_{sp} should be large enough to contain an appreciable about of texture, and small enough so that the state of strain inside the subset can be assumed to be homogeneous. Using a large value for M_{sp} provides more noise damping but may be inappropriate with large heterogeneity in strains.

2.7.3 Continuous Deformation Strain Tracking

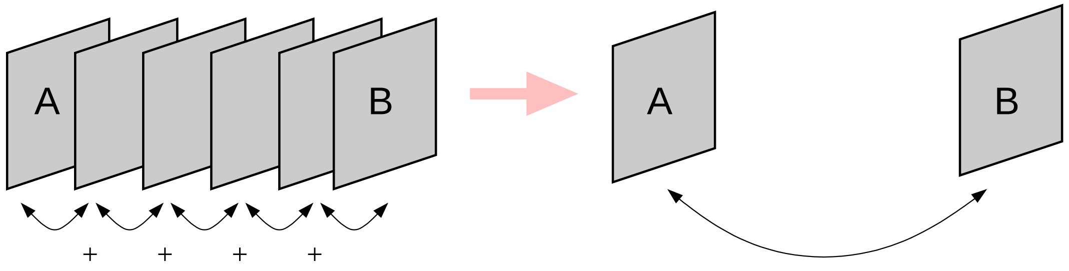


Figure 2.5: Image sequences corresponding to the loading cycles of multiple stretch protocols are tracked pair-wise and information passed sequentially to the next pair. Hence, the total deformation is broken down into incremental steps corresponding to the downsampled image sequence. Cumulative displacement fields are stored and Green strains computed from finite-element theory of bilinear quadrilateral elements.

The preceding sections describe the procedure for correlating two images where the accuracy of the DIC algorithms is dependent on small deformations between the images being correlated. In order to continuously track deformation corresponding to loading cycles of stretch tests, we perform the correlation process repeatedly on image-pairs drawn from an image sequence. This section briefly describes the passing of information between image pairs and the computation of cumulative displacements and strains.

The recorded video of tissue deformation is downsampled and processed to obtain a grayscale image sequence. The images are synchronized with the load cell data to identify the images corresponding to each loading protocol. The image of the sample before the start of test is used as the reference configuration and the region of interest is sketched and meshed with quadrilateral meshes as mentioned previously (Fig. 2.3). The meshes are mapped onto the reference image and successive pairs of images in the sequence are correlated (Fig. 2.5). For each image-pair (denoted as f_i and g_i), the locations of mesh nodes on f_i are rounded to the nearest integer pixel location and the corresponding point in g_i is identified via correlation algorithms described

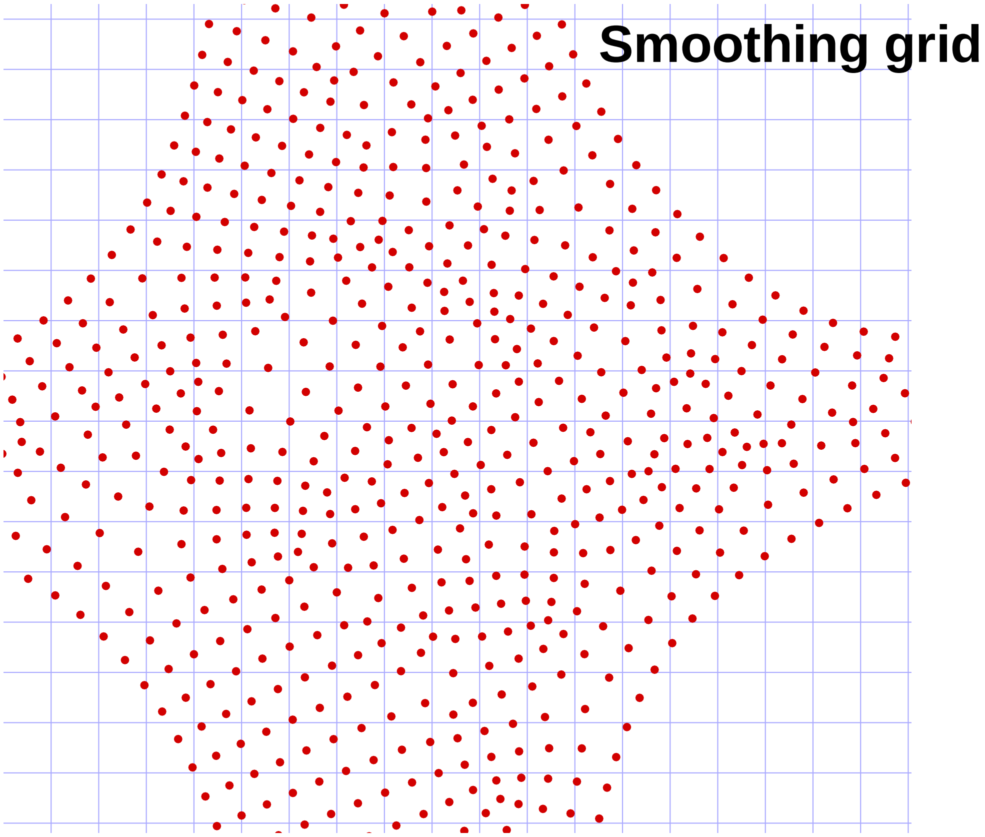


Figure 2.6: The displacement field obtained for every image-pair is smoothed before being passed to the next pair. The red nodes correspond to the rounded-off locations of the nodes in image f_i. Displacements at each node are treated as functional field values at the location of the scattered nodes. An oversized grid (shown in blue) is used to smooth the scattered function-data by a gradient smoothing operator (MATLAB$^{®}$ function `gridfit`). The values from the smoothed grid are interpolated onto the nodal locations in f_i, the corresponding displacements are added to the locations to identify the position of the nodes in image g_i, and passed to the next image-pair with $g_i = f_{i+1}$.

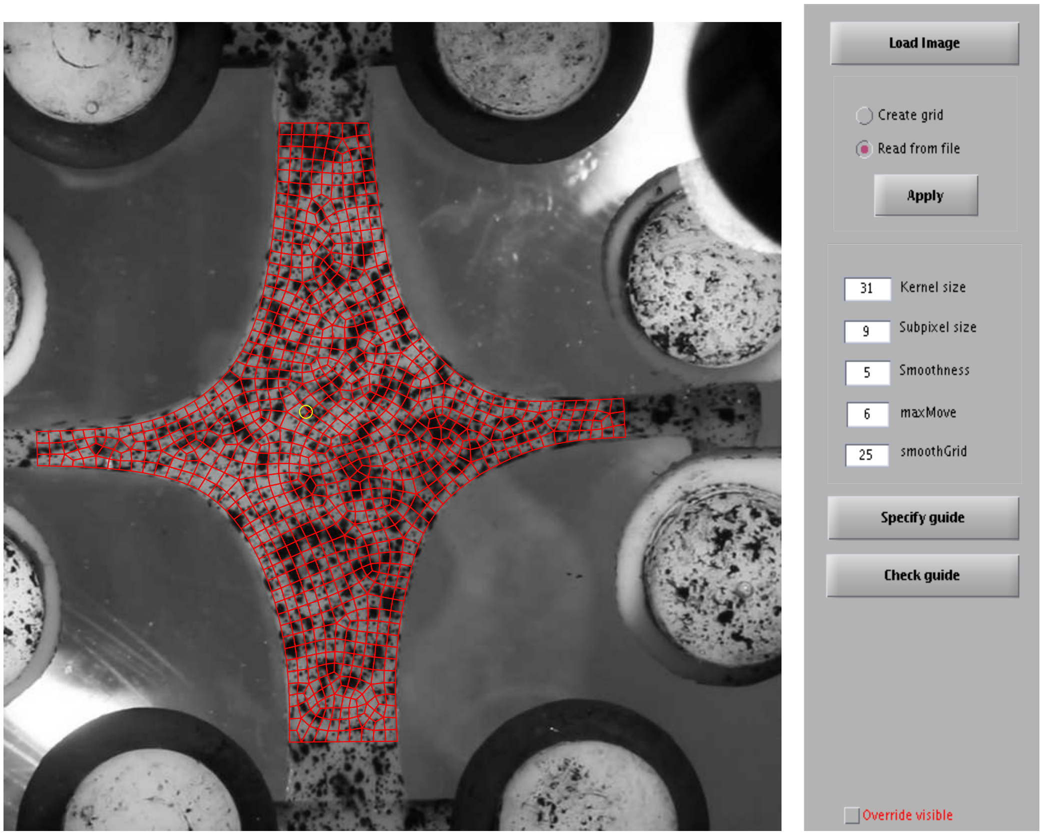
previously. We introduce a smoothing filter to dampen noise by transferring the nodal displacement values in x and y onto a grid that is slightly coarser than the fine-mesh. Figure 2.6 shows the rounded-off locations of nodes on f_i with a slightly oversized grid ($\approx$ 1.5–2 mesh edge lengths) for smoothing. The displacements in x and y are treated as functional field values at the location of scattered data, and smoothed using the

MATLAB® function `gridfit` which applies a gradient smoothing operation on the grid. The smoothed displacements from the grid are interpolated onto the nodes of both, the fine-mesh and coarse-mesh on f_i. The smoothed displacements are added to the nodal locations in f_i to obtain the corresponding nodal locations in g_i. This step is repeated for the next image-pair (f_{i+1}, g_{i+1}) where $f_{i+1} = g_i$, and the nodal locations directly passed between the two. Cumulative nodal displacements are stored for all steps for strain calculation. Strains are computed from the estimated nodal displacements using finite-element theory of bilinear quadrilateral mesh elements [43].

In general, the accuracy of displacement and strain tracking methods are dependent on a number of factors such as the quality of visual texture, lighting conditions, pixel interpolation functions and image resolution. Refs. [78,79] discuss these factors in detail. It is important to estimate the amount of uncertainty in strain specific to one's images by digitally deforming a representative image and comparing the computed displacements and strains with the analytical values.

2.7.4 Graphical User Interface for Strain Tracking

A graphical user interface (GUI) based program was developed in MATLAB® to perform the continuous deformation strain tracking. Snapshots of the GUI-based tracking are shown in Figs. 2.7,2.8. The user can read in the image sequence, load the fine-mesh, set the tracking parameters and specify the guide-node. The movement of the guide-node is estimated solely through ZNCC and the subset matches are shown in Fig. 2.8 together with the correlation surface. If the subset match is poor or if the peak of the correlation surface is not distinct, then the user can cancel the guide-node specification and change the tracking parameters accordingly.



○ Create grid
● Read from file

31 Kernel size
9 Subpixel size
5 Smoothness
6 maxMove
25 smoothGrid

□ Override visible

Figure 2.7: Graphical user interface for strain tracking: Finite-element mesh is overlaid on the image and tracked at nodal locations. Various tracking parameters can be specified; the user is asked to specify the guide-node which is chosen based on the region with best texture and least movement.

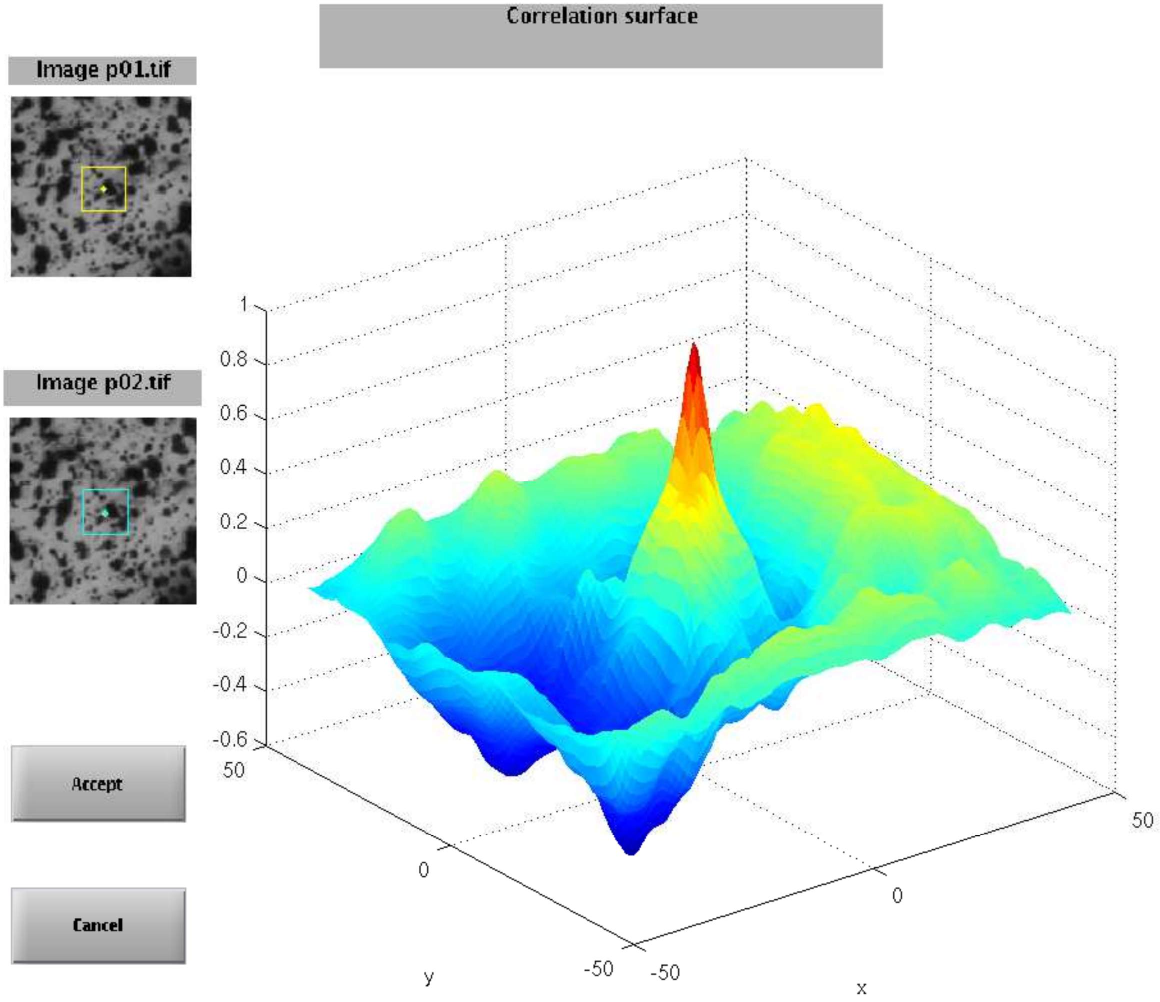


Figure 2.8: Graphical user interface for strain tracking: After specification of the guide-node, the subset match for that location is estimated by ZNCC, and the user asked to judge the matched regions for accuracy. Good match of the subsets in respective images (yellow and cyan boxes) and a distinct peak in the correlation surface indicate good performance for the chosen tracking parameters. A poor match in subsets, or the presence of large secondary peaks indicate that tracking is unrealiable for the current choice of parameters, and the user is asked to change them.