Nanoscale displacement measurement of microdevices via interpolation-based edge tracking of optical images

Assaf Ya’akobovitz, Davor Copic, Justin D Beroz and A John Hart

Mechanosynthesis Group, Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, USA
E-mail: assafy@umich.edu and ajohnh@umich.edu

Received 6 December 2012, in final form 27 January 2013
Published 18 February 2013
Online at stacks.iop.org/JMM/23/045004

Abstract
We apply an interpolation-based edge-tracking algorithm to measure nanoscale displacements of microdevices, and further enhance its ability to reject noisy signals using averaging of multiple image frames. We present a simulation of a moving edge, and use this simulation to explore the performance of the algorithm as related to the image acquisition and process parameters. Then, we present the application of this technique to two experiments. First, we demonstrate optical measurement of the motion of a compliant mechanism, and smoothly detect lateral step motion change as small as 30.8 nm s$^{-1}$. Second, we measure the anisotropic nanoscale expansion of a liquid crystal network micropillar, which occurs slowly over several minutes. This efficient algorithm can smoothly track motions as small as a few per cent of a pixel, which is equivalent to tens of nanometers using images from a video camera attached to a conventional optical microscope. Therefore, this technique can be widely applied to characterization of nanoscale motions using conventional optics, without requiring special features on the device to enable imaging.

(Some figures may appear in colour only in the online journal)

1. Introduction

Due to their non-destructive nature and relatively simple experimental setup, optical imaging methods have gained increasing popularity in the field of micro electromechanical systems (MEMS). The development of commercial software and image processing methods has contributed to the accessibility of these methods. Among the leading image processing algorithms are particle image velocimetry, which is widely used in the field of fluid mechanics [1], and digital image correlation (DIC), which is used in materials science and solid mechanics, such as to track strain fields within thin films under load [2, 3]. Edge tracking algorithms, which were first reported as early as 1971 [4], are attractive due to their combination of simplicity, robustness and performance.

In edge-tracking algorithms the location of a moving edge, which is usually indicated by an abrupt change in color intensity, is identified either based on a pre-defined color intensity threshold value, or using the gradient of intensity. In the most basic threshold-based edge-tracking algorithm, a line of pixels is considered; the location of the edge is determined to be the first pixel whose color intensity is at or below the pre-assigned threshold. In gradient-based methods, such as the well-known Canny algorithm [5], a threshold value for the intensity gradient is pre-defined and the edge is detected based on the gradient values that exceed this value. In both cases, by applying the algorithm to sequential images, the location of the edge is tracked with time.

Several studies have demonstrated that edge-tracking algorithms can achieve sub-pixel resolution. For example, these include the use of edge tracking for coarse shape tracking in conjunction with speckle tracking algorithm for finer edge tracking [6]. Another study used curve fitting of the edge intensity levels using a hyperbolic tangent function [7]. The interpolation-based edge-tracking algorithm demonstrated high-resolution measurement of the tilting of optical MEMS devices based on the position of a reflected laser beam [8, 9]. It has also been applied to track the

Nanoscale resolution optical measurements can be achieved using high-end experimental setups [12], for example using interferometry methods [13–15], laser Doppler vibrometer [16, 17] or confocal microscopy [18]. However, these methods require controlled laboratory conditions and pose many limitations on the experiment. In some cases, there is a limitation on the geometry of the analyzed devices, such as a need for accessible side walls [19], or a requirement for specific optical properties [20]. Alternatively, DIC-based image processing for nanoscale displacement measurement was published [21, 22], but in many cases the use of DIC is not possible, due to lack of the unique pattern required for this method or due to large displacements which challenge the DIC tracking. In addition, DIC processing often requires computational power greater than that needed for edge-tracking algorithms. Therefore, it remains an opportunity to extend edge-tracking algorithms to achieve nanoscale resolution as shown in this study, and to analyze the limits of edge-tracking performance based on the motion and image processing parameters.

In this paper, we demonstrate the use of interpolation-based edge-tracking to achieve smooth measurements of nanoscale displacements of both free-standing and surface-bound microscale objects. The performance of the algorithm is investigated as related to the operational parameters and the noise level. We apply this method to two experiments, which demonstrate nanoscale resolution measurements. First, we demonstrate optical measurement of the motion of an actuated metal edge, and detect motion change as small as 30.8 nm s⁻¹. Second, we measure the nanoscale anisotropic expansion of a liquid crystal network (LCN) micropillar.

2. Methodology: an averaged interpolation edge-tracking algorithm scheme

The edge-tracking algorithm used in this study is shown schematically in figure 1. We consider a line of pixels that passes through the edge. The algorithm extracts this line from the images, and searches along this line for pixels with color intensity levels that are below a pre-defined threshold value. From these pixels, the pixel with the maximal spatial coordinate is chosen. Next, the slope of a line connecting this pixel and the next pixel is calculated, and the intersection point between the threshold value and the interpolation line is defined as the location of the edge. Unlike the Canny algorithm [5], in which the gradient intensities and directions are first calculated for all pixels and then the edge is determined based on gradient thresholding, in our method the position of the edge is first determined coarsely based on the intensity threshold, and then its position is refined using the slope (gradient) to obtain sub-pixel resolution as indicated by the placement of the edge in the schematic.

Next, in order to further enhance the performance of the algorithm, images are averaged. The resulting displacements from the edge-tracking algorithm are divided into groups of $n$ images and each group is averaged. As will be shown below, the averaging step reduces the influence of measurement and processing noise. However, averaging is possible only in cases where images are acquired significantly faster compared to the velocity of the moving edge. A moving average was found to provide similar performance to a group average in the cases presented in this paper. However, when the edge velocity is higher, a moving average will provide better performance.

3. Analytical model

In order to evaluate the performance of the averaged interpolation-based edge-tracking algorithm, we first simulated analysis of a moving edge with variable noise. This enabled understanding of how the algorithm performance depends on the processing parameters and the noise level.

An ideal edge profile can be physically described as a step function. However, due to the transfer function of the optics, the image of the edge will not be a perfect step [7, 23, 24]. We describe the image of the edge as the convolution of a step function and a Gaussian function, which is equal to the following cumulative distribution function:

$$y(x) = \frac{1}{2} \left( 1 + \text{erf} \left( \frac{x - x_{\text{edge}}}{\sqrt{2} \sigma} \right) \right) \cdot 100 + 50 + \eta. \quad (1)$$

Here, erf is the error function, $x_{\text{edge}}$ is the location of the edge and $\sigma$ is the Gaussian blur [25], also known as blur strength [23]. The variable $\eta$ is referred to as the noise level and denotes a random noise which can be introduced to the edge function (see the discussion below). The manipulation of the standard function expressed in equation (1), specifically, multiplying by 100 and adding 50, sets the color intensity level of the function from 60 to 180.
Then, the color intensity detected by the \(i\)th pixel can be derived as

\[
I_{x_i} = \int_{x_i-FF}^{x_i+FF} y(x) \, dx. \tag{2}
\]

Here, \(x_i\) is the coordinate of the center of the \(i\)th pixel and \(FF\) is the fill factor (\(0 < FF \leq 1\)) which represents the effective part of sensitivity area of the pixel (as described in [7]). We chose a Gaussian blur of \(\sigma = 1\) and fill factor of \(FF = 60\%\). These represent a typical quality camera and similar parameters were used by Cantatore et al [7]. We normally took the number of averaged images as \(n = 50\). Equation (2) was calculated numerically using the trapezoidal rule.

In figure 2(a), we show the simulated color intensity level obtained for solution of equation (2), for an edge that moves from an arbitrary coordinate of 0 to 15. We also show the influence of noise on the performance of the edge algorithm. Specifically, a noise, which changes randomly between color intensity levels of \(-40\) to \(40\), namely \(\eta = 80\), in coordinate and in time (frame number), was added to the ideal edge function, figure 2(b). This noise can represent a wide range of noise sources, such as thermal noise of the moving body [26], systematic effect of an optical sensor [25], fluctuation of the light source, or environmental vibrations.

A plot of the color intensity levels versus the spatial coordinate is shown in figure 2(c) for zero displacement of
the edge; this represents the first image in the simulated sequence. The edge is bounded by an envelope (shown as the blue dashed line) and as a result the intensity value falls above or below a band, except at the location of the edge. This is also demonstrated in figure 2(a) for the same case as a function of the frame number. The existence of this band depends on the optical and physical parameters of the experiment. Nevertheless, when such a band is present, the threshold value should be chosen to fall within this band, enabling identification of the edge position with high accuracy (figure 2(c)).

Next, before considering the influence of averaging, the basic interpolation-based edge-tracking algorithm was used to extract the position of the edge whose motion was simulated (figure 2). The simulated position, as determined by the algorithm, is shown in figure 3. In the case of no noise the measured displacement looks smooth (figure 2(a)), but the introduction of noise causes the results to have significant scatter. Moreover, even without noise the algorithm predicts a scalloped line. As a result of the way the intensity is calculated, namely integration over the edge function, equation (2), the slope in which the refined location of the edge is based on slightly changes as the edge advances. When the real location of the edge moves to the next pixel, the discontinuity is observed. This represents the computational error of the algorithm and it is periodic, so that every time the coarse edge detection moves to the next pixel along the line, this error zeroes and it again builds with the motion. However, this error is typically significantly smaller than real experimental noise, and therefore is insignificant to the final results.

Next, the results of figures 3(a) and (b) were further refined using the averaging step \( n = 50 \), see figure 3(c). It is clear that both cases (with and without noise) are in good agreement with the prescribed (given) edge displacement (shown as the green line), which demonstrates that implementing the method under discussion is useful in filtering the introduced random noise.

In order to enhance our understanding of the algorithm, we performed a parametric study of how the processing parameters influenced the simulated edge motion. Due to the randomness of the noise signal, different simulations may give different numerical results for identical algorithm parameters; however, here we discuss general trends which can be deduced regardless of the noise level. In what follows, the root-mean-square (RMS) error was calculated for each simulation, and is defined as

\[
\text{RMS err} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_{\text{nom}}^i - x_{\text{calc}}^i)^2},
\]

(3)

where \( x_{\text{nom}}^i \) and \( x_{\text{calc}}^i \) are the nominal (given) and calculated (processed) edge displacements of the \( i \)th image, respectively.

First, the influence of the size of the averaged group, \( n \), was evaluated. Edge functions with random noise were generated for selected noise levels \( (\eta = 0, 5, 50, 80) \) and then were processed using different values of group size \( n \), as shown in figure 4(a). In all the cases, the error drops significantly as the group size increases from 1 to approximately 10, and, it may be expected that, considering the edge motion is linear, the error should decrease with further averaging (larger \( n \)). However, in some cases, for example for noise levels of \( \eta = 0, 5, 80 \), a certain optimum value of group size, referred to as \( n_{\text{opt}} \), is associated with minimum error. For \( n > n_{\text{opt}} \) the error increases, which implies that the averaged group size is too big for low error tracking of the edge. Note that as a result of the randomness of the noise, the value of \( n_{\text{opt}} \) varies and different simulations can result in different values of \( n_{\text{opt}} \). In addition, because of the random nature of the noise signal, the
high noise level is not always associated with higher error; for example, in figure 4(a) noise of $\eta = 50$ corresponds to higher error in the simulated tracking results than the case of $\eta = 80$. Also, due to the computational error (causing scalloping of the predicted line, as discussed earlier) there is an error even when the specified noise level is zero. Aside from the noise level and type, the details of the image such as the edge shape and the motion characteristics will determine the ideal group size, but this can be determined empirically by post-processing.

Next, the effect of the noise level was analyzed. Edge functions with random noise levels ranging from $\eta = 0$ to 80 were generated, and then processed using group sizes of $n = 2, 10, 50$ and 100, as shown in figure 4(b). Although affected by the randomness of the noise, in most cases smaller values of $n$ are associated with higher errors. Although the random nature of the sampled distribution gives higher error for certain cases (e.g., $\eta = 35$), the RMS error does not increase systematically with the noise level. This is one of the most important advantages of the algorithm under discussion, and makes its use favorable over other methods that are highly sensitive to the noise level.

4. Results

In this section, we demonstrate the successful implementation of the new algorithm to two experiments involving controlled nanoscale displacement of edges. Importantly, these experiments present real cases where, as in past studies [24, 23, 7], we find that an ideal edge shape is not necessarily achieved due to the geometry, lighting and materials involved in the experiment.

4.1. Measurement of a microstructure moving at constant velocity

In this experiment, a piezoelectric stepper motor was used to apply a periodic step motion of $30 \text{ nm s}^{-1}$ (1 step s$^{-1}$) to the stage of a compliant mechanism, as shown schematically in figure 5(a). The moving edge was imaged using a microscope (Zeiss AXIO Imager.A1m) and a CMOS camera (INFINITY1) with $800 \times 600$ pixels resolution (physical length of a pixel is 420 nm), at a sampling rate of $4.6 \text{ frames s}^{-1}$ (0.21 seconds per frame). A typical image of the moving edge is shown in figure 5(b).

In figure 6(a) the color intensity as a function of the frame number and coordinate is shown; the shape of the intensity profile indicates the apparent profile of the edge. Based on these data, a threshold intensity value of 70 was selected, and the algorithm was processed along with the maximum sampling rate (4.6 frames s$^{-1}$), giving the displacement trend shown in figure 6(b). Here, we see that the motion is linear; however, due to noise the line has significant spread of approximately $\pm 1$ pixel, corresponding to $\pm 420$ nm.

Results after averaging ($n = 50$) are shown in figure 7(a), for different sampling rates. Specifically, every 5th, 10th and 20th points of figure 6(b) were taken to achieve sampling rates of 0.92, 0.46 and 0.23 frame s$^{-1}$, respectively. This way, the results shown in figure 7 correspond to the same experiment
Figure 6. Processing results for the moving edge experiment, with a threshold intensity value of 70 and the sampling rate of 4.6 frame s$^{-1}$. (a) Measured color intensity levels versus the coordinate and the frame number. (b) Displacement obtained using the basic interpolation-based edge-tracking algorithm.

Figure 7. Processing results for the moving edge experiment after averaging at different sampling rates. (a) Measured displacement. The prescribed motor displacement is shown as a solid line. (b) Standard deviation of the position values within each averaged group.

(hence with the same displacement and noise), with only a different sampling rate.

In figure 6, the blue circles correspond to the measured displacement obtained for the sampling rate of 4.6 frames s$^{-1}$. Processed displacement for sampling rates of 4.6, 0.92 and 0.46 frame s$^{-1}$ is in good agreement with each other and with the prescribed displacement shown as linear solid line, while significant deviation is observed for 0.23 frame s$^{-1}$. The standard deviation of displacements for the averaged group, which provides an estimation of how noisy the measurement is, is shown in figure 7(b). Note that for the sampling rates of 4.6, 0.92 and 0.46 frame s$^{-1}$ the standard deviation is 0.1 pixel, which is significantly smaller than the measured results (measured displacements are of the order of several pixels). However, for 0.23 frame s$^{-1}$, higher standard deviation was obtained. Therefore, the error is invariant with the sampling rate above a certain threshold, which for the cases examined here is 0.23 frame s$^{-1}$.

The slope of the linear fit, which indicates the sensitivity of the measurement, is 0.0082 pixel/frame (30.8 nm s$^{-1}$) for sampling rates of 4.6 frame s$^{-1}$, or 0.0083 pixel/frame (31.2 nm s$^{-1}$) for sampling rates of 0.92 and 0.46 frame s$^{-1}$. Both of these estimates are very close to the velocity applied by the motor (30 nm s$^{-1}$). On the other hand, for the sampling rate of 0.23 frame s$^{-1}$, a slope of 0.0068 pixel/frame (25.5 nm s$^{-1}$) was calculated, which is 15% less than the known velocity.

4.2. Optical actuation of a liquid crystal polymer microstructure

Second, we demonstrate the use of the algorithm to monitor the optically induced expansion of a liquid crystal network (LCN) micropillar. The ability to undergo light-driven deformation makes LCNs attractive for use in microactuators and dynamic surfaces, since it allows remote actuation of the material. However, the precise quantification of the strains in these polymers, without requiring surface modification or attachment of mechanical devices, has been challenging.

For this experiment, LCN micropillars were casted from a polydimethylsiloxane (PDMS) mold. The mold was filled with a mixture of optically active and thermally active liquid crystal (LC) diacrylate monomers. The mixture was composed of 20 wt% optically active 4,40-bis[6-(acryloxy)hexyloxy]azobenzene monomers (BEAM Co.), 78 wt% thermally active RMM491 monomer mixture (EMD Millipore) and 2 wt% I-784 photoinitiator (Ciba) [27]. The LC monomers were aligned and cross-linked in a magnetic field at 75 °C thus forming a network [28]. Figure 8(a) shows a
Upon the exposure to polarized blue light, the azobenzene moieties in the LCN undergo trans-cis isomerization, reorienting parts of the LCN. The network reorientation is associated with shrinkage along the LC director, and an expansion orthogonal to it.

The cast LCN was imaged optically, figure 8(b), and lines of pixels from the upper, lower, left and right edges of the square LCN were analyzed using the edge-tracking algorithm. Unlike the first experiment, in which the edge was very clear due to the abrupt change in color (figure 5(b)), in this experiment the color intensity levels on the two sides of the edge are relatively close to each other. In addition, the displacements here are significantly smaller (less than one pixel over the course of the experiment).

Analysis of a line of pixels perpendicular to the top edge of the micropillar is shown in figure 9; the other edges had similar characteristics. The color intensity levels as a function of the frame number and coordinate are shown in figure 9(a), demonstrating a shape very different from the ideal shape of an edge. The distribution of the red color intensity levels for the top edge is shown in figure 9(b). A sparse intensity band is clearly noted, as indicated by the red dashed lines, therefore indicating the presence of the edge. Accordingly, the threshold value chosen for the processing of this experiment was 215, which falls near the center of the band. Analysis of the green color channel gave similar results. However, because the micropillar was actuated using blue light, the intensity of the source dominated the blue channel and complicated the appearance of the edge.

The displacement of the top edge obtained using the algorithm without averaging is shown in figure 9(c). The displacements measured for each edge, with averaging ($n = 50$), are shown in figure 10(a). The standard deviations, shown in figure 10(b), are of the same order as the captured displacement, indicating the noisy character of this experiment. Nevertheless, the algorithm measured the anisotropic shape change of the microstructure, which contracts in the lateral (horizontal on the page) direction (strain of about $-0.1\%$ was measured), and expands in the vertical direction (strain of about $0.25\%$ was measured). Previous work
using macroscopic samples of a similar LCN measured 0.15% photogenerated strain using a dynamic mechanical analyzer [29]; however, to our knowledge this is the first attempt to quantify the anisotropic mechanical behavior of the LCN within individual micropillars.

5. Conclusions

We presented an algorithm for interpolation-based edge tracking of optical images, allowing robust measurement of nanoscale displacements from optical images of microscale devices. First, a simulation was built using a mathematical model of an ideal edge in order to demonstrate the implementation of the algorithm under different conditions of noise and analyze the influence of the operational parameters. It was found that the averaging significantly improves the capability of the algorithm to reject both measurement noise and computational errors. Moreover, it was found that choice of the group size for averaging, \( n_{\text{avg}} \), results in an optimal (minimal) error. Note that although the algorithm was implemented in this study on linear or weakly nonlinear displacements, its implementation on more complicated functions is possible as well. As was previously demonstrated, the implementation of the basic algorithm does not depend on the shape of the tracked function [9, 10], while as shown above, the quality of the averaging depends on different operational parameters. We also note that while the current analysis was carried out on a single line of pixels, an arbitrary number of lines could be monitored either \textit{in situ} or \textit{ex situ}; thus the algorithm could be used to detect more complicated shape changes and motion paths.

The application of the technique to two experiments was presented. In the first experiment, a controlled nanoscale motion was generated by moving a suspended compliant mechanism. The algorithm demonstrated high-resolution measurement when a displacement sensitivity as low as or 30.8 nm s\(^{-1}\) was captured and demonstrated excellent agreement with the specified motor velocity. Next, the edge motion of an optically actuated cast LCN was recorded and processed. In this experiment which was characterized with high noise and subtle deflections, the algorithm successfully quantified the nanoscale anisotropic expansion and contraction of a polymer micropillar over timescale of several minutes.

The algorithm is therefore attractive due to its high resolution, good noise filtering, low sensitivity to the noise level and high computational efficiency. Therefore, implementation of this algorithm can aid in high performance characterization of motion in the development of microsystems and active materials.

Acknowledgments

We thank Professor Slava Krylov and Professor Samantha Daly for helpful discussions, and Chen Zur for her advice and interest in this study. Funding was provided by the Department of Energy Office of Basic Sciences (DE-SC0004927).

References