Real-time Rigid-body Visual Tracking in a Scanning Electron Microscope

Bradley E. Kratochvil, Lixin Dong and Bradley J. Nelson

The International Journal of Robotics Research 2009; 28; 498
DOI: 10.1177/0278364908099849

Bradley E. Kratochvil
Lixin Dong
Bradley J. Nelson
Institute of Robotics and Intelligent Systems,
ETH Zurich,
Tannenstrasse 3,
Switzerland
bkratochvil@ethz.ch

Real-time Rigid-body Visual Tracking in a Scanning Electron Microscope

Abstract

Robotics continues to provide researchers with an increasing ability to interact with objects at the nanoscale. As microrobotic and nanorobotic technologies mature, more interest is given to computer-assisted or automated approaches to manipulation. Although actuators are currently available that enable displacement resolutions in the subnanometer range, improvements in feedback technologies have not kept pace. Thus, many actuators that are capable of performing nanometer displacements are limited in automated tasks by the lack of suitable feedback mechanisms. This paper proposes the use of a rigid-model-based method for end-effector tracking in a scanning electron microscope to aid in enabling more precise automated manipulations and measurements. These models allow the system to leverage domain-specific knowledge to improve performance in a challenging tracking environment.

KEY WORDS—nanomanipulation, model-based visual tracking, scanning electron microscope.

1. Introduction

Robotic micromanipulation is changing the way microelectromechanical systems (MEMSs) are designed and fabricated, as well as how cellular and molecular biological structures are handled. Recent efforts in manipulating nanometer-sized structures such as carbon nanotubes and nanocoils have shown great promise in characterizing nanoelectromechanical systems (NEMSs) (Zhang et al. 2006). In order to enable researchers to perform more kinematically complex manipulations, and to help free them from the burden of performing every task manually, better manipulation strategies are needed.

The International Journal of Robotics Research
Vol. 28, No. 4, April 2009, pp. 498–511
DOI: 10.1177/0278364908099849

Figures 1, 3–12 appear in color online: http://ijr.sagepub.com

Manipulations are often performed in conjunction with devices such as optical microscopes, scanning electron microscopes (SEMs) or transmission electron microscopes (TEMs), which all provide sensory feedback in the form of a monocular image. The SEM is often favored for nanomanipulation owing to its high resolution and depth of field. Thus far, most nanomanipulations have focused on manual strategies for characterizing nanostructures (Weir et al. 2005), but recent work has explored the use of template matching (Sievers and Fatikow 2006) and active contours (Fatikow et al. 2007) for semi-automated visual servoing tasks.

One major challenge of performing visual servoing tasks inside a SEM is balancing the needs of image quality with real-time imaging. The sequential scanning used to create a SEM image necessitates lower frame rates than those available with optical cameras. SEM imaging quality is also highly dependent on the scale and material properties of the area being viewed. Often these parameters cannot be changed because they are related directly to the task being performed. Thus, methodologies must be developed to work with the limited frame rates and imaging restrictions of the SEM.

Current tracking algorithms can roughly be partitioned into feature-based, model-based methods and hybrid methods. Feature-based methods generally encompass methods such as active contours (Kass et al. 1988), affine covariant region detectors (Mikolajczyk et al. 2005) or techniques for detecting geometric primitives (Illingworth and Kittler 1988). Model-based methods capitalize on pre-existing knowledge of the tracking problem such as the geometrical structure of the target (Drummond and Cipolla 2002) or its interaction with the environment (Greminger and Nelson 2004). Hybrid methods incorporate aspects of both (Pressigout and Marchand 2007) with the desire to be robust to factors which would cause problems to methods employing only a single modality.

Although electron microscopes provide a host of new challenges to imaging systems, they also allow for some simplifying assumptions to be valid in real-world applications. In particular, manipulation tasks in a SEM generally have struc-

tured environments without a large number of occlusions or distractors. Many of the objects present in a manipulation task are either modeled previously or can be modeled easily due to the microfabricated nature of their construction. These facts have led us to choose rigid-body methods based on the work of Drummond and Cipolla (2002) and Yesin and Nelson (2005) to help provide more precise motion feedback for manipulation and measurement tasks. The model-based approach allows us to exploit knowledge of the task at hand to increase tracking precision and frame rate.

We first analyze the imaging process of a SEM, and how it can be used for computer vision applications. We then discuss how the use of rigid models allows the user to incorporate task specific constraints into the system to improve tracking performance and stability. This system can be easily reconfigured for two- or three-dimensional tracking roles. The structure of the tracking problem can additionally be analyzed for portions of the observation which have more leverage on the final tracking solution. These areas can be scanned selectively, thus alleviating one of the principle challenges to real-time operation in the SEM: the frame rate. The performance of this methodology is then analyzed through tracking a microgripper under a variety of conditions.

2. SEM as a Vision Sensor

It is often remarked that given the significant difference between how optical and SEM images are generated, the latter are easily interpreted by even novice users. This is due in part to the fact that the image is mapped similarly to a traditional optical system. A typical SEM image is generated by rastering a stream of electrons across the sample of interest. The beam generated at the electron gun generally has a spot size too large to produce a sharp image, and thus a series of electron lenses must be used to focus the beam.

The beam is generated at the electron gun which accelerates the electrons to an energy in the range of 0.1–30 keV. The beam then passes through a series of condenser lenses, which reduces the spot size. Lowering the beam size increases the resolution of the system, but also increases the signal noise owing to the decrease in energy being returned from the sample. The scan coils are then used to raster the beam across the image, as opposed to an optical image sensor, which acquires all of the data in parallel. Typically, the sample time for each pixel ranges in the order of 50–250 ns, where multiple samples at individual pixel locations are used to increase the signal-to-noise ratio (Figure 2). Finally, the beam passes through an objective lens, which focuses on the observed object (Goldstein et al. 2003).

When the electron beam scans across the material of interest, a number of signals are generated. Of primary concern to computer vision applications are the secondary electrons (SEs) and backscattered electrons (BSEs). SEs are loosely bound

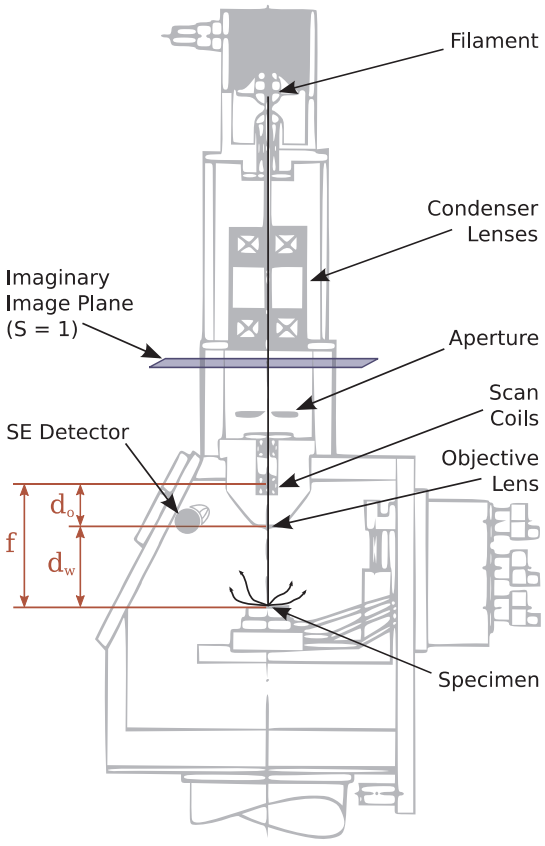
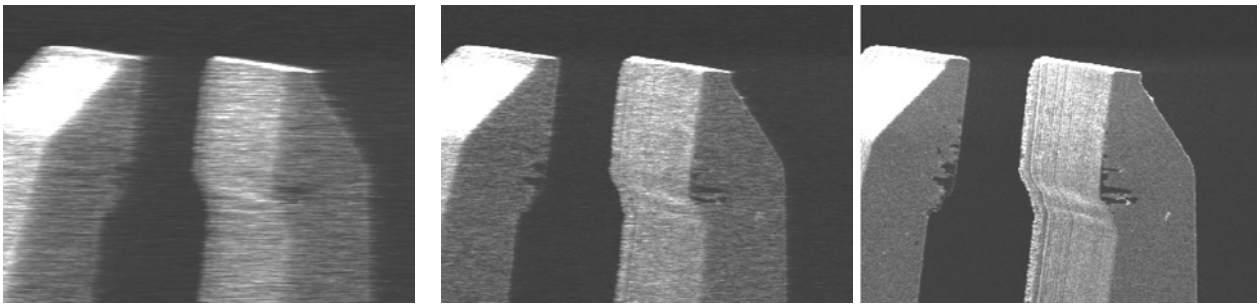


Fig. 1. Schematic of an electron column, where f is the focal length of the system. The working distance d_w is provided by the SEM during a measurement, but the distance between the objective lens and scan coils d_o must be found through calibration.

outer-shell electrons that receive enough kinetic energy to be ejected by the inelastic scattering of the beam electrons. These are typically low-energy (less that 50 eV) electrons and provide a diffuse image of the sample. BSEs are high-energy electrons that are generated by elastic scattering and generate a more specular image. Although different detectors are used to sense each signal, the electron optics remain the same for each image type, which enables us to use the same model for different imaging modalities.

Analogous to optical systems, the electromagnetic lenses in these systems are also subject to irregularities that can lead to spatial and time-varying distortions. Sutton et al. (2006) examines these phenomena for metrology applications and found that, although the spatial distortion is relatively small across the image (approximately 0.5 pixels maximum), the drift ranges up to 20 pixels over 2 hours at the magnifications they used. These noise parameters are within an acceptable range for visual servoing tasks.



(a) 16 Hz (b) 4 Hz (c) 0.5 Hz

Fig. 2. SEM images of a microgripper at a different frame rates.

Aside from resolution, electron microscopes have a significant advantage over optical microscopes in terms of depth-of-field. Since objects remain in focus at distances farther from the focal plane, a perspective model should be used for relatively low magnifications (less than 1,000×), whereas a parallel projection model is valid for higher magnifications (Cornille et al. 2003).

2.1. Perspective Projection

Using the standard pinhole camera model, a point in the world coordinate frame can be mapped to the SEM image plane as

$$\begin{bmatrix} U \\ V \\ S \end{bmatrix} = \underbrace{\begin{bmatrix} f_u & s & u_0 & 0 \\ 0 & f_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{P} E_{4\times 4} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{1}$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \frac{U}{S} \\ \frac{V}{S} \end{bmatrix}, \tag{2}$$

where $[u \quad v]^{\mathrm{T}}$ is a point on the normalized $S = 1$ image plane, and $[x \quad y \quad z \quad 1]^{\mathrm{T}}$ is a point in the world frame, expressed in homogeneous coordinates. The matrix P is known as the set of *intrinsic parameters* of the imaging system. The matrix E is a homogeneous transformation matrix of the *extrinsic parameters*, which relate the camera's coordinate frame to the world coordinate frame. Using this relationship, we are now able to relate points in image space to vectors in world space (Faugeras 1993).

2.2. Parallel Projection

As the distance from the object to the image plane increases, the z distance between features becomes relatively small when compared with the viewing distance. When this occurs the image approaches an orthographic projection and the projection matrix for the SEM can be a simple scaling factor k and a skew s such that

$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \begin{bmatrix} k & s & 0 & 0 \\ 0 & k & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}. \tag{3}$$

For magnifications greater than 1,000× this assumption can be made because the effective size of the pixel becomes more than the shift in the image plane due to the perspective projection (Cornille et al. 2003).

3. Rigid-body Tracking

Our tracker is inspired by the trackers developed by Bregler and Malik (1998), Drummond and Cipolla (2002) and Yesin and Nelson (2005). The model pose with respect to the camera is stored in the matrix $H \in \mathrm{SE}(3)$ such that a model point $\mathbf{x}$ is projected into image space by

$$\begin{bmatrix} U \\ V \\ S \end{bmatrix} = PH\mathbf{x}. \tag{4}$$

The special Euclidean group SE(3) is commonly know in the robotics literature as homogeneous transformations. The Lie

algebra of SE(3), denoted se(3), is identified by a 4 × 4 skew symmetric matrix of the form:

$$\begin{bmatrix} 0 & -\omega_3 & \omega_2 & v_1 \\ \omega_3 & 0 & -\omega_1 & v_2 \\ -\omega_2 & \omega_1 & 0 & v_3 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \hat{\xi}. \tag{5}$$

The mapping from se(3) to SE(3) is performed by the exponential formula $H = e^{\hat{\xi}}$ and a closed-form solution exists through the Rodriguez formula. We refer to the matrix $\hat{\xi}$ as a *twist*. Similar to Murray et al. (1994), we define the $\vee$ (vee) operator to extract the six-dimensional *twist coordinates* which parametrize a twist,

$$\begin{bmatrix} 0 & -\omega_3 & \omega_2 & v_1 \\ \omega_3 & 0 & -\omega_1 & v_2 \\ -\omega_2 & \omega_1 & 0 & v_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}^{\vee} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \xi. \tag{6}$$

The motion between consecutive frames can be represented by right multiplication of H with a motion matrix M. The generators of the group SE(3) can be defined as the translations in the x, y and z directions, as well as rotations about their respective axes expressed in the current model frame. The *twist coordinates* for these generators are

$$g_i = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \tag{7}$$

These generators form a basis for the vector space of derivatives of SE(3) at the identity and use the exponential map to obtain group elements such that

$$M = e^{\sum_{i=1}^{6} \alpha_i \hat{g}_i}. \tag{8}$$

The values of α are scalar quantities such that $\alpha = [\alpha_1 \ldots \alpha_6]^{\mathrm{T}}$. Assuming that the motion between frames is small, a linear approximation can be made for M using Taylor's theorem:

$$M \approx I + \sum_{i=1}^{6} \alpha_i \hat{g}_i. \tag{9}$$

Since the motion is approximately a linear sum of each of the generators, the partial derivatives for each point $\mathbf{x}$ with respect to each generating motion can be computed as

$$\begin{bmatrix} U_i' \\ V_i' \\ S_i' \end{bmatrix} \approx P H \hat{g}_i \mathbf{x} \tag{10}$$

$$L_i = \begin{bmatrix} \dfrac{\partial U_i}{\partial \alpha_i} \\ \dfrac{\partial V_i}{\partial \alpha_i} \end{bmatrix} = \begin{bmatrix} \dfrac{U_i'}{S} - \dfrac{U S_i'}{S^2} \\ \dfrac{V_i'}{S} - \dfrac{V S_i'}{S^2} \end{bmatrix} \tag{11}$$

3.1. Edge Tracking

The algorithm leverages the aperture problem, which states that only motion normal to an edge can be detected. Thus, the edge-tracking phase can be simplified to searching for intensity gradients along a one-dimensional path normal to a point on the edge. The first stage is to determine which edges are currently visible in the view. The model is provided for the system in the form of a boundary representation model that can be drawn in commercially available CAD packages. These models consist of triangular meshes that cover the surface of the model. An initial preprocessing step can be used to classify these edges into either "invisible", "silhouette" or "sharp". Invisible edges occur when a flat surface is converted into a mesh of triangles and cannot be viewed. Sharp edges are generated by two faces of the original model coming into contact. Silhouette edges are edges that are generated along the face of a curved surface, and are only capable of being viewed when they make up part of the silhouette of the image. The use of triangular meshes allows for arbitrary shapes to be modeled using this method.

The system initially renders the model using estimates for the object pose H. These estimates can be supplied by the user or another image localization package. Using a method such as z-buffering or binary space partition (BSP) trees, a user can determine which edges of the polygon mesh are visible. These model edges are then divided into m small segments for tracking. Once the midpoints are determined and the unit-normal $\mathbf{n}^{\psi}$ constructed, where ψ denotes the ψth segment being analyzed, a line of l points is searched for the edge location. This is performed by convolving the search line with a one-dimensional kernel $\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}$ and analyzing the resulting values. Although we currently use a gradient kernel, operators such as the Laplacian can also be used for edge detection.

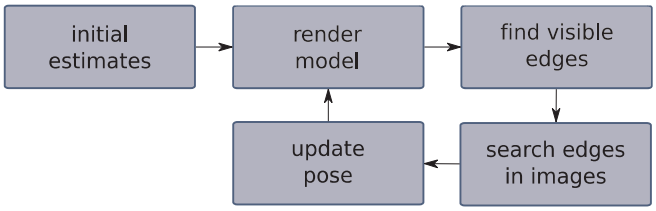


Fig. 3. Flowchart of system operation

The distance from the midpoint to the detected edge is then recorded as d_ψ. Other useful information can also be recorded in this step for use with heuristics to improving tracking. More information on this topic is provided in Section 3.3.

3.2. Pose Estimation

The final step is pose estimation. With knowledge of both the edge normal and the partial derivatives with respect to generating motions, we are able to build an $m \times 6$ image Jacobian that describes the normal component of observed motion with respect to the ith generating motion as

$$J_{\psi i} = \frac{\partial d_\psi}{\partial \alpha_i} = \mathbf{n}^\psi \cdot L_i^\psi \tag{12}$$

such that

$$J\boldsymbol{\alpha} = \mathbf{d}, \tag{13}$$

where $\mathbf{d}$ is the $m \times 1$ vector of d_ψ values. Formulating the problem in this fashion allows us to use a least-squares approach to find the updated pose. If J has full column rank, then $J^{\mathrm{T}}J$ is invertible and the following equation can be used to solve for α:

$$\boldsymbol{\alpha} = (J^{\mathrm{T}}J)^{-1}J^{\mathrm{T}}\mathbf{d}. \tag{14}$$

In the case that J does not have full column rank, methods such as those based on singular value decomposition can be utilized to find the best estimate of α.

Once the new update of α is available, the pose of the tracked object is updated and used as the input for the tracking cycle.

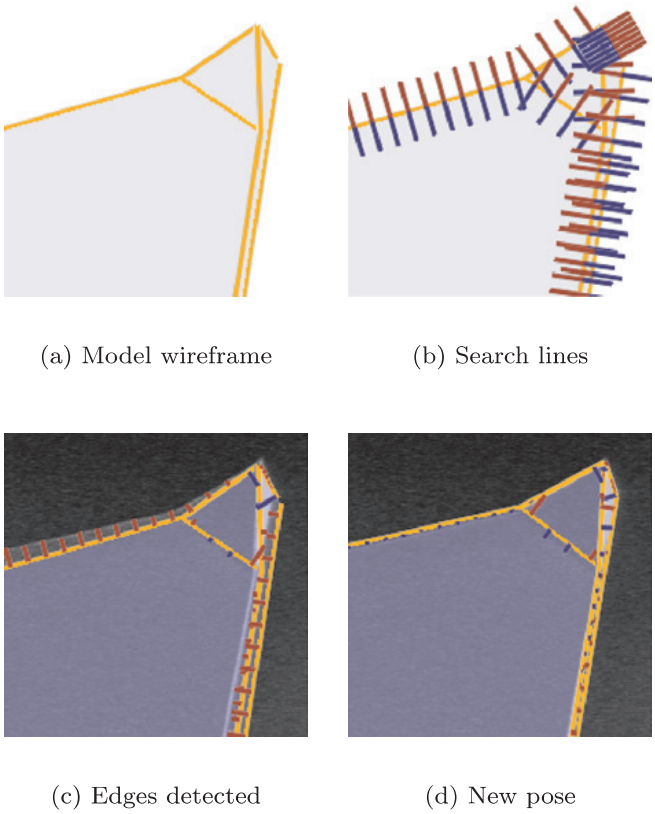
$$H_{t+1} = H_t M. \tag{15}$$

The flow of the algorithm in illustrated in Figures 3 and 4.

3.3. Row Weighting

The large number of samples helps to reduce the influence of outliers in the least-squares estimation, but additional steps can be taken to improve the quality of the fitting. If extra information is known about particular samples, this data can be used to weight the influence of a particular measurement.

$$\begin{aligned} WJ\boldsymbol{\alpha} &= W\mathbf{d}, \\ W &= \mathrm{diag}(w_1, w_2, \ldots, w_m). \end{aligned} \tag{16}$$



(a) Model wireframe (b) Search lines

(c) Edges detected (d) New pose

Fig. 4. The images display one iteration of the pose estimation sequence: (a) the hidden edges are removed, (b) the normal lines are searched, (c) the edges are located and (d) a new pose is estimated.

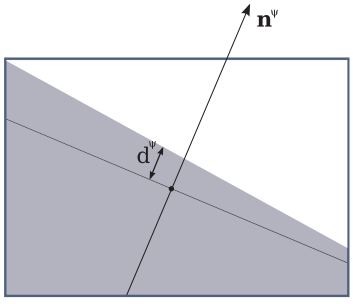


Fig. 5. Error measurement along the normal direction.

This scaling transforms the problem into an iterative least-squares problem (Golub and Loan 1996). The iterations can either occur at each new image frame, or owing to the relatively slow refresh rate of the SEM can occur multiple times on a single image. Drummond and Cipolla (2002) suggest a variety of different methods for choosing these weighting functions which attempt to solve different domain-specific prob-

lems. These criteria and others can be multiplicatively combined to provide an estimate for w_ψ.

- *False matches*. Since least-squares assumes a Gaussian distribution, outliers in the edge detection step can cause a large change in the estimated pose. To alleviate this problem the weighting function can be adjusted to

$$w_\psi = \frac{1}{c + |d_\psi|}. \tag{17}$$

  A good value for the constant c is one standard deviation of the error measure.

- *Multiple edges*. If multiple edges are on the search line, the tracker is capable of choosing the wrong one. This problem can be reduced by making the weight inversely proportional to the number of edges detected on the search line.

- *Edge quality*. The edge strength can be cited as a helpful metric because the weaker edges are harder to track. A set of two thresholds can be used to generate the weights. Below a certain threshold, no edge is detected. Between the two thresholds the weight varies linearly, and above a threshold the weight is not adjusted.

3.4. Constrained Tracking

The tracking performance can be improved if domain-specific information can be factored into the pose estimation process. One of the simplest ways to incorporate this extra domain knowledge is by reformulating the problem to perform a least-squares minimization with equality or inequality constraints:

$$\begin{aligned} J\alpha &= \mathbf{d}, \\ C\alpha + \beta &\leq \mathbf{0}, \\ C_{\text{eq}}\alpha &= \mathbf{0}. \end{aligned} \tag{18}$$

A variety of efficient software libraries such as LAPACK and Matlab exist for solving these types of problems. To better illustrate how these constraints are built, we briefly discuss a few examples.

3.4.1. Planar Constraint Example

The simplest case is to put a constraint on tracking a single object. For example, this could be due to the object resting on a ground plane. In this case, we would like to restrict translations along the z-axis and not allow rotations around the x- and y-axes of the object being tracked.

$$c^{1,2,3} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}^{\mathrm{T}}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}^{\mathrm{T}}. \tag{19}$$

We can write the constraint equations as

$$\begin{bmatrix} c^1 \\ c^2 \\ c^3 \end{bmatrix} \alpha = \begin{bmatrix} \mathbf{0}_{3\times 1} \end{bmatrix} \tag{20}$$

and solve the constrained least-squares optimization problem.

3.4.2. Optical Axis Constraint Example

Owing to the projection mode of the SEM, the image data contains minimal depth information. The absence of this information can cause instabilities in the fitting operation. As shown in Section 4, it is beneficial to restrict motion along the optical axis with constraints in the camera frame such as

$$c^1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \tag{21}$$

Since the pose is estimated in the body frame, the constraint must be transformed from the camera frame. The adjoint operator provides a convenient method for transforming a twist from one coordinate frame to another (Murray et al. 1994). Given $M \in \mathrm{SE}(3)$, the adjoint transform is a 6×6 matrix which transforms twists from one coordinate frame to another:

$$M = \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{22}$$

$$\mathrm{Ad}(M) = \begin{bmatrix} R & \widehat{\mathbf{t}}R \\ \mathbf{0}_{3\times 3} & R \end{bmatrix}. \tag{23}$$

The adjoint operator is invertible, and is given by

$$\mathrm{Ad}^{-1}(M) = \begin{bmatrix} R^{\mathrm{T}} & -R^{\mathrm{T}}\widehat{\mathbf{t}} \\ \mathbf{0}_{3\times 3} & R^{\mathrm{T}} \end{bmatrix}. \tag{24}$$

The adjoint operator can thus be used to express the constraints in the camera frame as

$$c^1 \mathrm{Ad}(H)\alpha = 0. \tag{25}$$

3.4.3. Multiple Object Constraint Example

Although multiple objects can be tracked independently, often one also wishes to track articulated objects. We can first build a sparse matrix that includes the Jacobians of both objects to be tracked.

$$\begin{bmatrix} J_1 & \mathbf{0}_{m\times 6} \\ \mathbf{0}_{m\times 6} & J_2 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \end{bmatrix}. \tag{26}$$

In the case of multiple objects each of their points are defined in their local frames. If we decide to choose the constraints in the local frame of α_1, then we must transform the motions of α_2 to the frame of α_1 for constraint checking. Since the transformation from the coordinate frame of α_2 to α_1 is described by the homogeneous matrix $H_1^{-1}H_2$, we can state

$$c_1^i\alpha_1 + c_2^i \mathrm{Ad}(H_1^{-1}H_2)\alpha_2 = 0 \tag{27}$$

and thus

$$\begin{bmatrix} c_1^1 & c_2^1\mathrm{Ad}(H_1^{-1}H_2) \\ \vdots & \vdots \\ c_1^\varsigma & c_2^\varsigma \mathrm{Ad}(H_1^{-1}H_2) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{\varsigma\times 1} \end{bmatrix}. \tag{28}$$

The vectors $c^{1\cdots\varsigma}$ are then defined in the frame of α_1. Since we would like the combined motion of α_1 and α_2 along the constrained axes to be zero, the constraints can be described as

$$c_2^i = -c_1^i. \tag{29}$$

In the case of tracking the arms of a microgripper, we would like to restrict the motion between the gripper arms to planar motions and ensure that no rotations are allowed between the bodies. Thus, planar constraints similar to Equation (19) can be used for c_1.

3.5. Tracking in SE(2)

The previous examples were provided for tracking objects with three-dimensional coordinates. Two-dimensional tracking is useful for objects that will not rotate out of plane during a manipulation as well as objects that are not precisely modeled prior to the experiment. In the latter case, the rigid body model can automatically be created from an image of the object taken at the beginning of an experiment or through methods such as active contours. The algorithm discussed previously can be analogously applied to tracking planar objects in SE(2). An element in se(2) is represented as

$$\begin{bmatrix} 0 & -\omega_1 & v_1 \\ \omega_1 & 0 & v_2 \\ 0 & 0 & 0 \end{bmatrix}^\vee = \begin{bmatrix} v_1 \\ v_2 \\ \omega_1 \end{bmatrix} \tag{30}$$

Given $M \in \mathrm{SE}(2)$, the adjoint transform is the 3×3 matrix

$$\mathrm{Ad}(M) = \begin{bmatrix} R & \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\mathbf{t} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix}. \tag{31}$$

The adjoint operator is invertible, and is given by

$$\mathrm{Ad}^{-1}(M) = \begin{bmatrix} R^{\mathrm{T}} & -R^{\mathrm{T}}\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\mathbf{t} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix} \tag{32}$$

In this case we use parallel projection, and our image projection becomes

$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \begin{bmatrix} k & s & 0 \\ 0 & k & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{33}$$

and the generating motions are

$$g_i = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{34}$$

Since there is no dependence on Z, the partial derivatives with respect to the generating motions are simply

$$L_i = \begin{bmatrix} U_i' \\ V_i' \end{bmatrix}. \tag{35}$$

All other operations can be applied in the same fashion as outlined above. Selection between tracking in SE(2) or SE(3) depends on the projection model selected as well as the desire to track out-of-plane motions. Often manipulators used at the microscale and nanoscale are constructed entirely of prismatic joints, making the SE(2) tracking method an attractive alternative to three-dimensional tracking.

3.6. Region-of-Interest Tracking

As shown in Table 1, the image size has a significant influence on the achievable frame rates of any tracking system used in the microscope. It becomes desirable then to reduce the number of pixels sampled while retaining the information content of the image. The image generation process of the SEM lends itself to selectively scanning different regions of interest.

Table 1. The Frame Rate versus Image Size and Sample Time of the DISS-5 Imaging System (Examples of Corresponding Images are given in Figure 2).

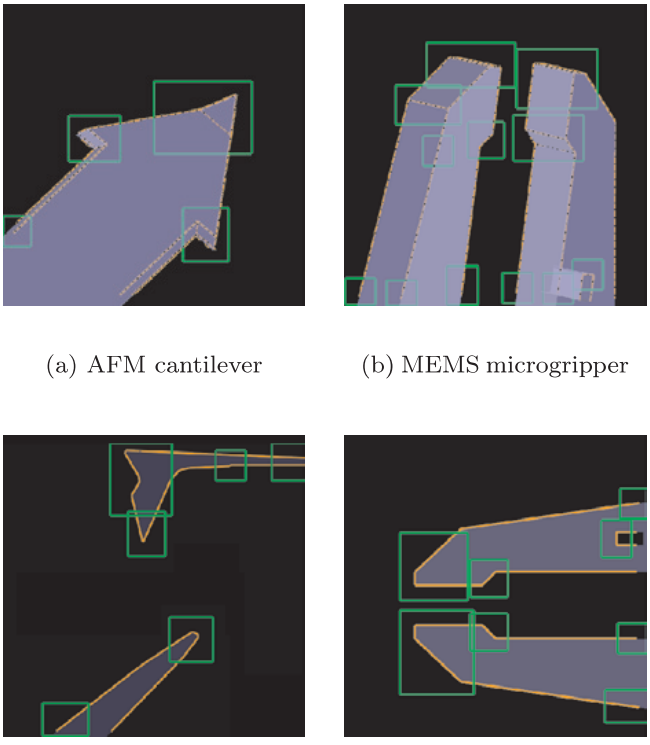
Pixel	Image size			
time	100 × 100	200 × 200	500 × 500	1,000 × 1,000
250 ns	400 Hz	100 Hz	16 Hz	4 Hz
500 ns	200 Hz	50 Hz	8 Hz	2 Hz
1,000 ns	100 Hz	25 Hz	4 Hz	1 Hz
2,000 ns	50 Hz	12.5 Hz	2 Hz	0.5 Hz
4,000 ns	25 Hz	6.25 Hz	1 Hz	0.25 Hz

These regions could be statically chosen at the beginning of an experiment or preferably dynamically adjusted at runtime to fit the task at hand.

In the case where J has full column rank, the least-squares solution to the pose estimation can be solved using (14). In this case, the vector of fitted values $\hat{\mathbf{d}}$ corresponding to the observed values $\mathbf{d}$ is

$$\hat{\mathbf{d}} = J\hat{\boldsymbol{\alpha}} = J(J^{\mathrm{T}}J)^{-1}J^{\mathrm{T}}\mathbf{d}. \tag{36}$$

The $m \times m$ matrix $H = J(J^{\mathrm{T}}J)^{-1}J^{\mathrm{T}}$ is often called the hat matrix, and maps the vector of observed values into a vector of fitted values (Montgomery 2005). The hat matrix can be used to identify influential observations based on their effect on the fitting operation. The elements h_{ij} can be interpreted as the amount of leverage exerted by d_j on $\hat{d}_i$. By inspecting the values of H, we can determine points that are more influential owing to their spatial position. We focus on the diagonal elements H_{ii}, which have an average value of n/m, where $n = 6$ for SE(3). Heuristically, we can select any point that is greater than $\lambda n/m$, where λ is a user-chosen threshold, as a point of high importance and use it as the center of a region of interest. Alternatively, a target frame rate can be set and λ dynamically chosen such that a given frame rate is achieved. Since multiple points in a region of the tracked object may meet this criteria, the selected points are clustered to reduce the amount of area to be scanned multiple times. The results of the automatic region-of-interest selection are shown in Figure 6 for multiple tracking objects. This automatic region selection is then added into the imaging loop. After each pose estimation, a new set of regions is dynamically calculated and subsequently sampled from the SEM.



(a) AFM cantilever (b) MEMS microgripper

(c) AFM cantilever and probe (d) MEMS microgripper

Fig. 6. Automatically selected regions of interest with a threshold $\lambda = 1.2$.

4. Experimental Results

The following experiments were performed with a Zeiss DSM 962 SEM. The manufacturer states that this microscope has approximately 10 nm resolution. The image acquisition is handled by a DISS-5 scan controller by Point Electronic GmbH. In the subsequent examples, an image size of 500 × 500 was selected to allow for a large viewing area and the possibility for sampling at a real-time rate (from 4 to 16 Hz).

A MEMS microgripper by Femtotools GmbH (shown in Figure 7) and an atomic force microscopy (AFM) cantilever have been selected as the tracking targets and will be used as the baseline objects for our tracking evaluation. These devices allow for tracking experiments at a variety of magnification factors. The microgripper also provides an opportunity to apply the constraints discussed in Section 3.4.

Table 2. The Standard Deviation of a Single Gripper Arm in Constrained and Unconstrained Tracking Modes with a Pixel Size of 0.229 μ**m.**

	σ_x	σ_y	σ_z	σ_ϕ	σ_β	σ_γ
Unconstrained	0.102 μm	0.279 μm	40.686 μm	0.002 rad	0.009 rad	0.001 rad
Multiple object	0.094 μm	0.081 μm	28.105 μm	0.002 rad	0.001 rad	0.001 rad
Optical axis	0.076 μm	0.047 μm	0.002 μm	0.004 rad	0.001 rad	0.001 rad
Multiple object + optical axis	0.035 μm	0.037 μm	0.000 μm	0.004 rad	0.001 rad	0.001 rad

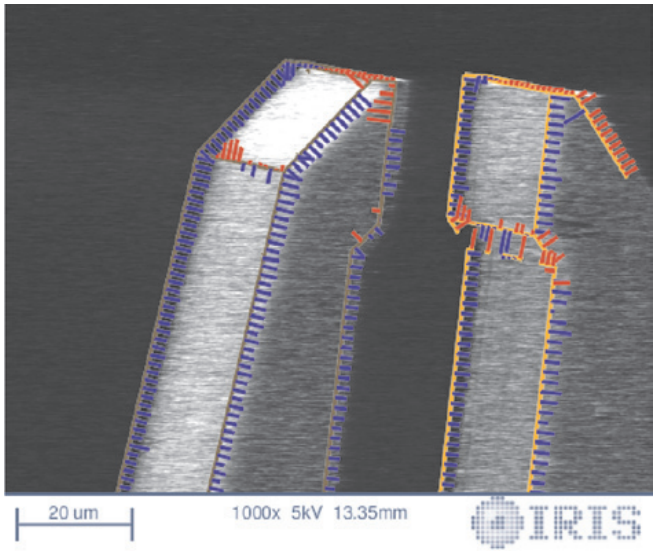


Fig. 7. SEM image of a microgripper being tracked with model overlay.

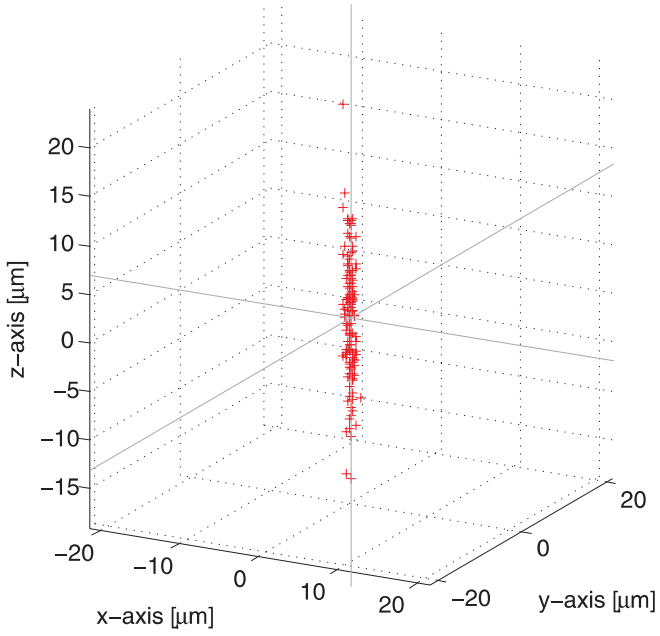


Fig. 8. The position estimates from 100 iterations of a static microgripper at magnification 500× and pixel sample time 500 ns. The error along the z-axis coincides with the optical axis of the SEM and is the most prone to noise.

4.1. Performance in SE(3)

The algorithm's precision in $SE(3)$ was determined by tracking a static target over a series of 100 frames and observing the variance of the resulting poses. In the following experiments the search length was set to 16 pixels, the multiple image and edge strength saliency metrics enabled, and in addition to the false matches metric the measurements with the largest 5% of displacement were classified as noise and removed from the fitting process. Unless otherwise mentioned, the data presented is for images acquired with a pixel sampling time of 500 ns or, equivalently, 4 Hz.

The uncertainty of the pose is determined by both the point matching errors as well as the directional dependent sensitivity of the system. As expected, Figure 8 clearly shows that the tracker's precision is significantly decreased along the optical axis of the system.

The influence of using constraints on precision was determined by tracking one gripper arm in the same sequence of images both with and without constraints. The constraints between the gripper arms were defined as described in Section 3.4.3. When running the constrained tracker, the distance between points was increased so that the total number of points being tracked matched that of a single arm at approximately 480. The results are shown in Table 2 and show a significant improvement in x- and y-axis precision when the constraints for z-axis motion in the camera frame are applied. The residual motion in the z-axis is a result of the numerical nature of the constrained estimation process. An additional improvement is shown when both gripper arms are used in the tracking process. This is most likely due to the increased separation of feature points caused by the addition of the second gripper. These additional points cause the entire estimation problem to be mathematically better conditioned than when the points are physically close. Owing to the improved performance

Table 3. The Standard Deviation of the Tracked Gripper Pose at Different Magnifications.

	Pixel size	σ_x	σ_y	σ_z	σ_ϕ	σ_β	σ_γ
200×	1.143 μm	0.112 μm	0.475 μm	0.004 μm	0.008 rad	0.025 rad	0.000 rad
500×	0.547 μm	0.103 μm	0.056 μm	0.003 μm	0.010 rad	0.023 rad	0.001 rad
1,000×	0.229 μm	0.058 μm	0.034 μm	0.002 μm	0.004 rad	0.003 rad	0.001 rad

Table 4. The Standard Deviation of the Cartesian Coordinates versus the Number of Sampled Points with a Pixel Size of 0.229 μm.

Sampled points	σ_x	σ_y	σ_z
∼120	0.075 μm	0.063 μm	0.001 μm
∼240	0.055 μm	0.043 μm	0.001 μm
∼480	0.035 μm	0.037 μm	0.000 μm
∼960	0.036 μm	0.024 μm	0.000 μm

Table 5. The Standard Deviation of the Tracked Gripper Pose at Different Sampling Rates.

Sample rate	Magnification			
	250×		500×	
	σ_x	σ_y	σ_x	σ_y
16 Hz	0.950 μm	8.429 μm	0.682 μm	0.481 μm
8 Hz	0.180 μm	0.215 μm	0.243 μm	0.315 μm
4 Hz	0.076 μm	0.107 μm	0.093 μm	0.097 μm
2 Hz	0.106 μm	0.109 μm	0.056 μm	0.088 μm
1 Hz	0.051 μm	0.120 μm	0.021 μm	0.063 μm

of algorithm using constraints, the remainder of the experiments utilize both the multiple object and optical axis constraints.

The tracking precision at a number of different magnifications was analyzed and the results are given in Table 3. In every case, the standard deviations of the x and y positions were below the effective pixel size of the image. At all magnifications, the rotational variance around the x-, y- and z-axes (σ_ϕ, σ_β and σ_γ, respectively) remained similar. This is due to the fact that translational errors scale based on magnification, while rotational errors do not. It should also be noted that when switching between 200× and 500× magnification, there is a large decrease in the standard deviation along the y-direction. This highlights the influence of the object geometry on the tracking precision of the system. At 200× magnification, the geometry of the gripper was such that there were very few edges visible orthogonal, or nearly so, to the y-axis. This means that the tracking problem was mathematically ill-posed along this axis. With higher magnification, more edges were visible in this orientation and thus tracking performance improved.

The effect of the number of feature points on the pose estimation was analyzed as shown in Table 4. The performance improves with the number of feature points, but plateaus after a sufficient number is reached to dampen image noise. The rotational standard deviations are not shown, but all varied similarly between 0.001 and 0.003 rad. Table 5 demonstrates that sampling rate has a significant influence on the tracking precision. Similar to the number of feature points though, this influence tapers off and provides a useful metric for evaluating the minimum sampling time for tracking.

4.2. Performance in SE(2)

The algorithm's precision was determined by tracking a static target over a series of 100 frames and observing the variance of the resulting poses. In the following experiments the search length was set to 10 pixels, the multiple image and edge strength saliency metrics enabled and the measurements with the largest 2% of displacement were classified as noise and removed from the fitting process. The system was allowed to converge on a pose for 20 turns before data was stored. Figure 9 shows the subsequent results of tracking 100 different frames which resulted in standard deviations of 10.4 and 9.56 nm in the x- and y-directions, respectively. The pixel size of the images used was 222 nm, which indicates that the tracking precision was better than $1/20$th of a pixel.

Although the least-squares estimation step is unbiased, the fact that each new pose is estimated relative to the previous pose suggests that there could be some bias in the estimation process. To investigate this, a single image from the previous precision experiment was tracked with different starting poses. These positions were taken from a uniform distribution on the intervals ± 3 μm and ± 0.05 rad. Figure 10 shows the results of 20 different initial poses after 20 iterations. The resulting standard deviations of 8.3 and 7.6 nm are similar in magnitude to the precision of the system as mentioned previously, indicating that the tracking algorithm is not largely biased by the initial estimates.

It then becomes interesting to analyze the number of iterations until the system reaches its precision limit. Figure 11

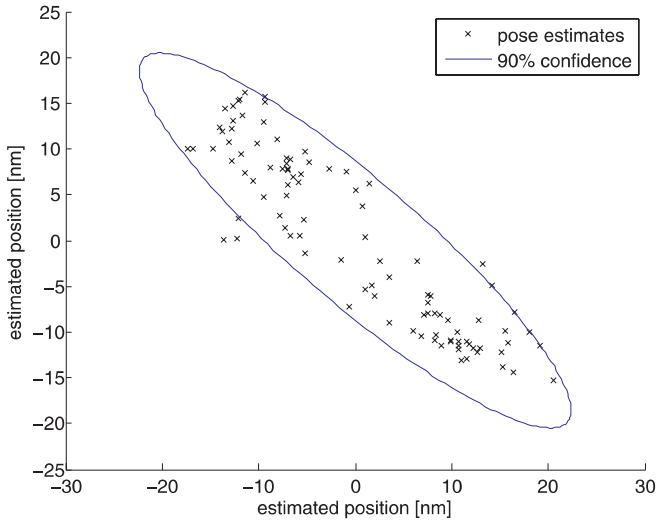


Fig. 9. Results of tracking a static AFM cantilever for 100 frames.

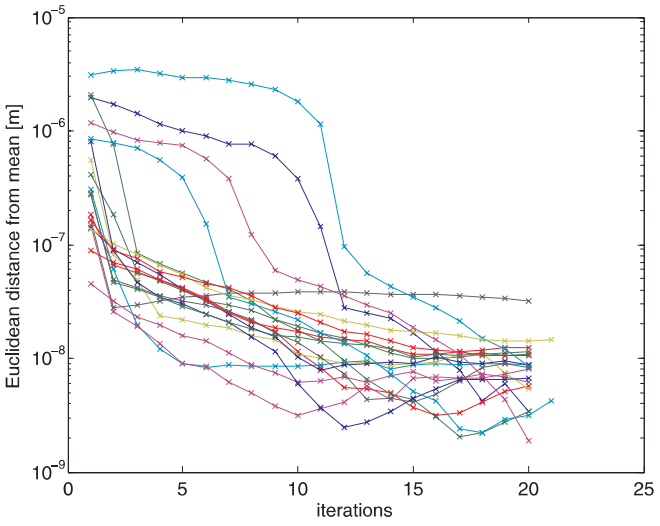


Fig. 11. Convergence of different starting poses over multiple iterations.

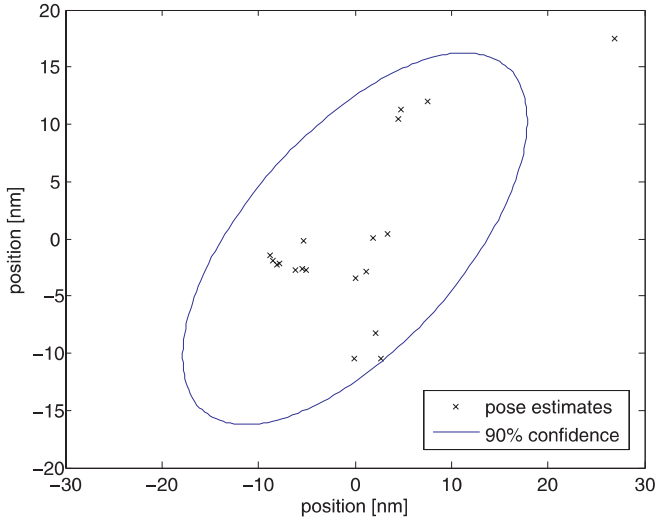


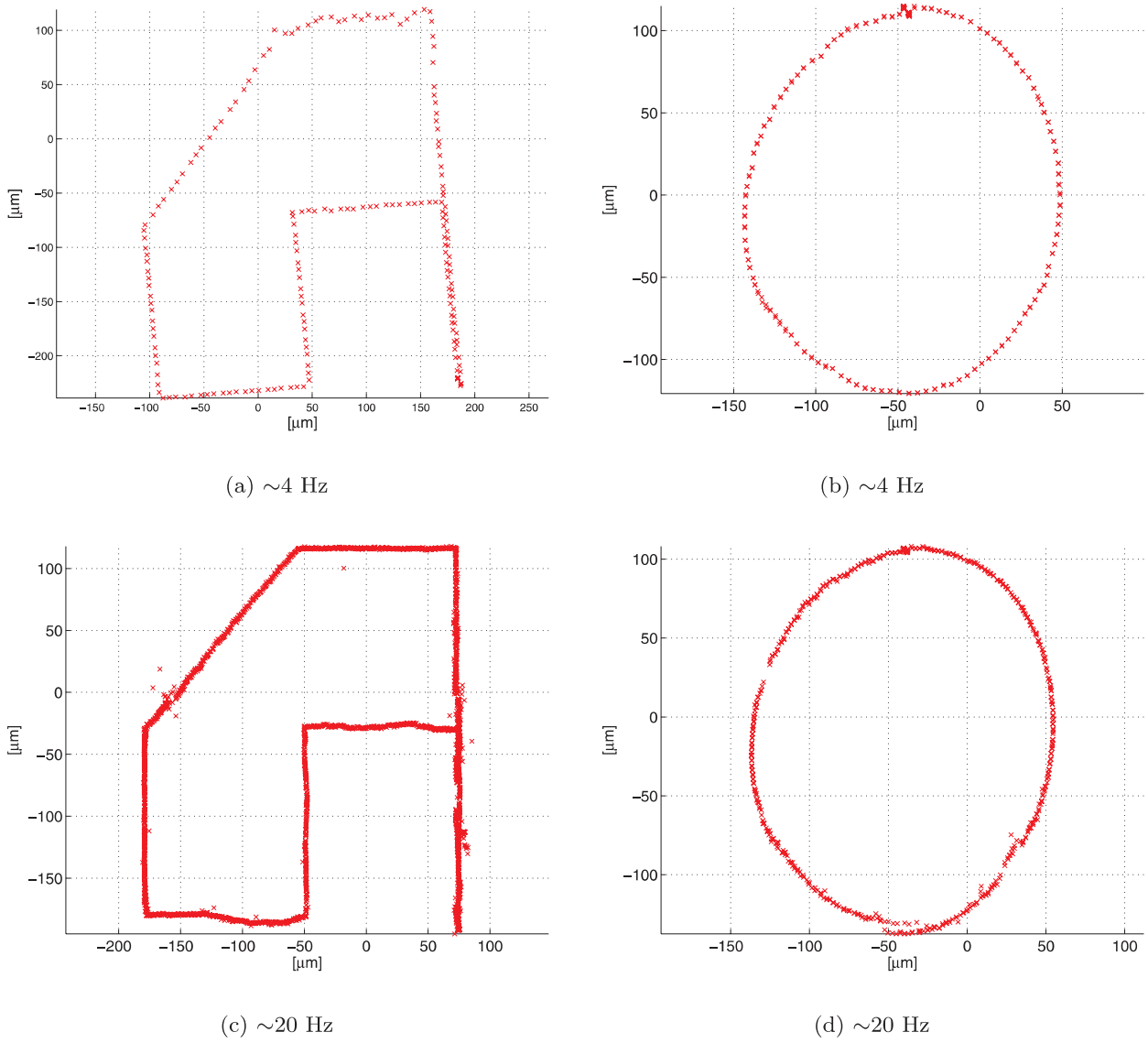
Fig. 10. Results of tracking a single image with different starting poses.

shows the convergence of the 20 different starting poses on their final estimates. One notable feature of this chart is that once the pose estimate approaches a neighborhood approximately equivalent to 2 pixels in image space from the final pose estimation, the algorithm only requires two or three iterations to approach the precision limit. This is partially a result of the linearization step in Equation (9) as well as the fact that the closer the estimate is to the actual pose, the larger number of edges that can be tracked. The initial poses in the experiment shown in Figure 11 were intentionally chosen to be a large distance from the true value. When tracking objects that only move a small number of pixels between subsequent frames, the algorithm can rapidly converge on the target value with as few as one or two iterations. Since the image refresh rate of the SEM for similar imaging operations is of the order of 10 Hz and the current tracking implementation performs in excess of 100 Hz with approximately 170 tracked points, as many as 10 iterations can be performed on each new SEM image which can aid in tracking faster moving objects.

4.3. Region-of-Interest Performance

To better understand the algorithms performance in both full-frame and region-of-interest tracking scenarios, two test trajectories were generated. The rectangle trajectory was intentionally designed with a variety of straight motions segments in different orientations to qualitatively evaluate the algorithm's performance. Although the actuators are driven in open-loop mode, the straight lines are generated by a single actuator, which allows us to better separate the possible actuator noise from that of the tracking algorithm. The circular pattern was chosen because it involves translational motions in all directions and should thus have limited bias based on the geometry of the tracked object. The following experiments were performed with the SE(2) tracker with a search length of ± 12 pixels along the edge normal and with the tracking constraints between the gripper arms applied.

Figure 12 shows sample trajectories tracked with the algorithm at 200× magnification. In all four cases shown, the algorithm successfully tracked the target for the entire run. Qualitatively, it can be seen in the rectangular path that there is more noise in the pose estimates along the y-axis than along that of the x-axis. This is primarily due to the orientation of



(a) ∼4 Hz (b) ∼4 Hz

(c) ∼20 Hz (d) ∼20 Hz

Fig. 12. Typical results of the trajectory tracking experiments. Trajectories (a) and (b) were tracked using a full 500 × 500 image, while (c) and (d) were tracked using automatically generated regions of interest. Note the higher density due to the increased frame rate of the region-of-interest poses.

the gripper during the experiment. As noted previously, the gripper geometry is such that there are significantly more feature points along the y-axis than the x-axis, which allows for a more precise measurement. While Figures 12(a) and (b) were tracked using the full sensor frame at approximately 4 Hz, Figures 12(c) and (d) were tracked using the region-of-interest strategy. Aside from significantly denser poses, the tracking precision of the two methods does not appear to differ appreciably.

A series of tests were then run on the algorithm using both the full-frame and region-of-interest strategies. In both cases, the gripper was tracked over the trajectory at a series of different velocities. The results are shown in Tables 6 and 7. As anticipated these results demonstrate that the region-of-interest strategy performs significantly better at high-speed tracking tasks than its full-frame equivalent due to the increased frame rate. During the course of these tracking tasks, the scan regions are dynamically generated at each frame. These regions were selected to provide an anticipated frame rate of 20 Hz based on the sample time at each pixel and the area of the scanned regions.

Table 6. The Success Rate of Tracking the Circle Trajectory at Different Velocities out of Five Trials Each.

Velocity		Full-frame (∼4 Hz)	Region of interest (∼20 Hz)
138 μm s^{-1}	122 pixels s^{-1}	0%	60%
84 μm s^{-1}	74 pixels s^{-1}	0%	100%
36 μm s^{-1}	32 pixels s^{-1}	100%	100%
28 μm s^{-1}	25 pixels s^{-1}	100%	100%

Table 7. The Success Rate of Tracking the Rectangle Trajectory at Different Velocities out of Five Trials Each.

Velocity		Full-frame (∼4 Hz)	Region of interest (∼20 Hz)
138 μm s^{-1}	122 pixels s^{-1}	0%	40%
84 μm s^{-1}	74 pixels s^{-1}	0%	100%
36 μm s^{-1}	32 pixels s^{-1}	100%	100%
28 μm s^{-1}	25 pixels s^{-1}	100%	100%

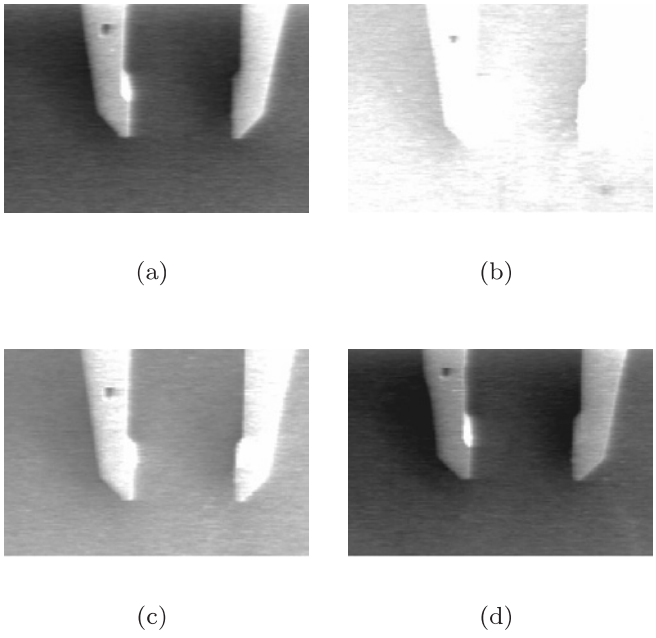


Fig. 13. Highlights of noisy images extracted from a rectangular tracking image series during region-of-interest operation. The system successfully dealt with all of these variations and tracked to the end of the experiment.

4.4. Imaging Noise

A variety of sources contribute to image noise when working with a SEM. Much of the speckle noise seen on an image is due to the nature of detecting the emitted electrons and can be reasonably modeled as Gaussian (Prasad and Joy 2003). This noise can often be directly mitigated through a reduction of the sampling rate of the image at the expense of the frame rate. Additional noise in the imaging pipeline is caused by charging effects as a manipulator interacts with the environment. This can take the shape of large intensity and contrast variations in the image. Figure 13 demonstrates a number of different effects of this charging taken from a single experimental run. Although the system successfully tracked during these abberations, they pose a serious challenge to the robustness of any visual system and effort should be devoted to reducing these effects. Even in the best cases, image noise and distortions cannot be entirely eliminated and future systems should consider hybrid feature and model-based methods (Pressigout and Marchand 2007) to provide added system stability.

5. Outlook and Conclusions

We have described and demonstrated a model-based tracking system suitable for use in a SEM under a variety of conditions. This system has demonstrated an ability to work with noisy images and sub-pixel resolution at a number of magnifications, and the ability to track two- and three-dimensional moving targets at real-time frame rates. The use of rigid-body models along with the incorporation of additional domain-specific knowledge, via constraints, provide mechanisms for improving system precision and robustness, which may not be available using purely feature-based methods. Using this methodology the structure of the observation at hand can be used to automatically select regions which are highly influential on the tracking tasks. These regions can then be selectively scanned with the electron beam to provide a reduced scan area and thus higher frame rates or higher imaging quality. The region-of-interest strategy has demonstrated the ability to track targets at up to five times that of a full-framed system.

Owing to the often low signal-to-noise ratio and possible image distortions, SEMs provide a challenging environment for real-time computer vision applications. Hybrid methods which combine both feature- and model-based information have shown an ability to robustly perform in a variety of different operating environments (Marchand et al. 2001; Pressigout and Marchand 2007), and could be of significant benefit in this domain. In addition to new imaging difficulties, SEM imaging also provides a number of interesting possibilities due to data acquisition differences from traditional optical systems. We have exploited one such difference in the application of our real-time region-of-interest strategy, but many others exist.

With low-level access to a SEM's scan controllers, a researcher is able to easily customize their imaging sensor to the task at hand.

In conclusion, although using SEMs for manipulation related tasks is a relatively young field, it provides many interesting opportunities for computer vision tools. Automated systems can help reduce the time required to perform measurements and manipulations at the nanoscale, which is crucial when both machine and user time is expensive. We have demonstrated a system which helps provide initial steps in this direction.

Acknowledgements

This work is conducted with financial support from the ETH Zurich and the project "Hybrid Ultra Precision Manufacturing Process Based on Positional- and Selfassembly for Complex Micro-Products (HYDROMEL)" funded by the European Commission under the 6th Framework Program (FP6).

References

Bregler, C. and Malik, J. (1998). Tracking people with twists and exponential maps. *Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 8–15.

Cornille, N., Garcia, D., Sutton, M. A., McNeill, S. and Orteu, J. J. (2003). Automated 3D reconstruction using a scanning electron microscope. *Proceedings of the SEM Conference on Experimental and Applied Mechanics*.

Drummond, T. and Cipolla, R. (2002). Real-time visual tracking of complex structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **24**(7): 932–946.

Fatikow, S., Wich, T., Hülsen, Sievers, T. and Jähnisch, M. (2007). Microrobot system for automatic nanohandling inside a scanning electron microscope. *IEEE/ASME Transactions on Mechatronics*, **12**(3): 244–252.

Faugeras, O. (1993). *Three-dimensional Computer Vision*. Cambrdige, MA, The MIT Press.

Goldstein, J., Newbury, D., Joy, D., Lyman, C., Echlin, P., Lifshin, E., Sawyer, L. and Michael, J. (2003). *Scanning Electron Microscopy and X-ray Microanalysis*, 3rd edition. New York, Springer.

Golub, G. H. and Loan, C. F. V. (1996). *Matrix Computations*, 3rd edition. Baltimore, MD, The Johns Hopkins University Press.

Greminger, M. A. and Nelson, B. J. (2004). Vision-based force measurement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**: 290–298.

Illingworth, J. and Kittler, J. (1988). A survey of the Hough transform. *Computer Vision, Graphics, and Image Processing*, **44**(1): 86–116.

Kass, M., Witkin, A. and Terzopoulos, D. (1988). Snakes: active contour models. *International Journal of Computer Vision*, **1**(4): 321–331.

Marchand, E., Bouthemy, P. and Chaumette, F. (2001). A 2D—3D model-based approach to real-time visual tracking. *Image and Vision Computing*, **19**(13): 941–955.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and Gool, L. V. (2005). A comparison of affine region detectors. *International Journal of Computer Vision*, **65**(1/2): 43–72.

Montgomery, D. C. (2005). *Design and Analysis of Experiments*, 6th edition. Hoboken, NJ, Wiley.

Murray, R. M., Li, Z. and Sastry, S. S. (1994). *A Mathematical Introduction to Robotic Manipulation*. Boca raton, FL, CRC Press.

Prasad, M. S. and Joy, D. C. (2003). Is SEM noise Gaussian. *Microscopy and Microanalysis*, **9**(2): 982–983.

Pressigout, M. and Marchand, E. (2007). Real-time hybrid tracking using edge and texture information. *The International Journal of Robotics Research*, **26**(7): 689–713.

Sievers, T. and Fatikow, S. (2006). Real-time object tracking for the robot-based nanohandling in a scanning electron microscope. *Journal of Micromechatronics*, **3**(3): 267–284.

Sutton, M., Li, N., Garcia, D., Cornille, N., Orteu, J. J., McNeill, S. R., Schreier, H. W. and Li, X. (2006). Metrology in a scanning electron microscope: theoretical developments and experimental validation. *Measurement Science and Technology*, **17**(10): 2613–2622.

Weir, N. A., Sierra, D. P. and Jones, J. F. (2005). A review of research in the field of nanorobotics. *Technical Report SAND2005-6808*, Sandia National Laboratories.

Yesin, K. B. and Nelson, B. J. (2005). A CAD model based tracking system for visually guided microassembly. *Robotica*, **23**: 409–418.

Zhang, L., Ruh, E., Grützmacher, D., Dong, L. X., Bell, D. J., Nelson, B. J. and Schönenberger, C. (2006). Anomalous coiling of SiGe/Si and SiGe/Si/Cr helical nanobelts. *Nano Letters*, **6**: 1311–1317.