

Developing robust vision modules for microsystems applications

Hakan Bilen · Muhammet A. Hocaoglu ·
Mustafa Unel · Asif Sabanovic

Received: 18 September 2009 / Revised: 28 December 2009 / Accepted: 1 April 2010
© Springer-Verlag 2010

Abstract In this work, several robust vision modules are developed and implemented for fully automated micromanipulation. These are autofocusing, object and end-effector detection, real-time tracking and optical system calibration modules. An image based visual servoing architecture and a path planning algorithm are also proposed based on the developed vision modules. Experimental results are provided to assess the performance of the proposed visual servoing approach in positioning and trajectory tracking tasks. Proposed path planning algorithm in conjunction with visual servoing imply successful micromanipulation tasks.

Keywords Microsystems · Micromanipulation · Visual feedback · Robust detection · Normalized cross correlation · Tracking · Visual servoing · Path planning

1 Introduction

In his famous lecture “There is plenty of the room at the bottom” in 1959, Richard Feynman considered a number of interesting ramifications on the problem of manipulating and controlling things on a small scale. In his talk Feynman described the problem of writing an enormous amount of text, the entire 24 volumes of the Encyclopedia Britannica,

on a surface about the size of the head of a pin. He also discussed even more interesting possibilities for constructing micromachines such as a tiny mechanical surgeon which can go into blood vessels. Although it was not possible to realize his big dream for small things at that time, he deserves the credit for recognizing the potential of miniaturization of the conventional devices that will have a major impact on society and everyday life.

With the recent advances in micro and nanotechnology, the commercial markets, including microelectromechanical system (MEMS) products such as the key components in automobile airbags, ink-jet printers and projection display systems, have been growing quite rapidly. Despite the fact that these commercially available microdevices are currently produced in a batch technique with little assembly, many other products such as read/write heads for hard disks and fiber optics assemblies require flexible precision assemblies. However, the assembly of these products are mostly done in manual or semi-automatic operations. Furthermore, many biological micromanipulations such as in vitro fertilization, cell characterization and treatment rely on the ability of human operators. Requirement of high-precision, repeatable and financially viable operations in these tasks has given rise to the elimination of direct human involvement, and autonomy in micromanipulation and microassembly.

In the literature, several research efforts on autonomous microrobotic tasks (micromanipulation and microassembly) under optical microscopes equipped with cameras can be found in [1–15]. Boissenin et al. [1] and Amavasai et al. [2] developed several customized machine vision methods (calibration, recognition, tracking and depth reconstruction) for closed-loop microrobotic control systems. Kim et al. [3] proposed a hybrid assembly method, which combines the vision-based microassembly and the scaled teleoperated microassembly with force feedback. With these tools, 980 nm

H. Bilen · M. A. Hocaoglu · M. Unel (✉) · A. Sabanovic
Faculty of Engineering and Natural Sciences, Sabanci University,
Istanbul, Turkey
e-mail: munel@sabanciuniv.edu

H. Bilen
e-mail: hakanbil@su.sabanciuniv.edu

M. A. Hocaoglu
e-mail: muhammet@su.sabanciuniv.edu

A. Sabanovic
e-mail: asif@sabanciuniv.edu

pump laser is manufactured by manipulating and assembling optoelectrical components. Beyeler et al. [8] have reported a robotic workstation consisting of an MEMS gripper and an ultrasonic manipulator. Manipulation experiments with polymer spheres, glass spheres and cancer cells are demonstrated using ultrasonic alignment and force feedback. Wang et al [15] present a microrobotic system for fully automated zebrafish embryo injection based on computer vision and motion control. The microrobotic system performs autonomous injection at a high speed, with a successful survival rate.

Different microscope technologies such as scanning electron microscope (SEM) [16] and atomic force microscope (AFM) [17] are also preferred to monitor the manipulation process in the literature. Although these monitoring technologies ensure higher resolutions, SEM can be used only in vacuum environments for some specific particles and AFM cannot provide an online imaging during manipulation tasks. Thus, optical microscope for manipulating microparticles has certain advantages including real-time imaging, reconfigurable workspace and low cost, despite the fact that small working distance and depth of field pose challenges to the three-dimensional micromanipulation and microassembly.

Automated assembly of micron-scale components is still an open and active research field. There exists strong demand for accurate and robust control approaches to compensate the inaccuracies in the mechanical design due to imprecise manufacturing, and to autonomously combine microparts from two or more different sources under non-clean room conditions. It is also desired to have more degrees of freedom for handling full 3D structures, instead of thin planar objects, with minimum user guidance and tuning.

This study aims to develop robust and provably correct generic algorithms based on feedback provided from the imaging sensors, which significantly enhance our ability to observe, position, and physically transform the microobjects and contribute to applications ranging from biotechnology to MEMS. First, several robust vision modules such as autofocus, feature detection (object and probe detection), real-time tracking and optical system calibration are developed and implemented. Second, a path planner is developed and integrated into the vision modules to enable the system to achieve fully automated micromanipulation. Based on the developed vision modules and the path planner, an optimal closed-loop control architecture is proposed to compensate errors, which arise in open loop control. By utilizing this controller, the versatility and accuracy of a microassembly workstation is greatly improved. Figure 1 depicts the system diagram with all the vision modules and relations between them and other system components. A fully automated micromanipulation task, which aims to arrange microspheres through a collision-free path using visual feedback

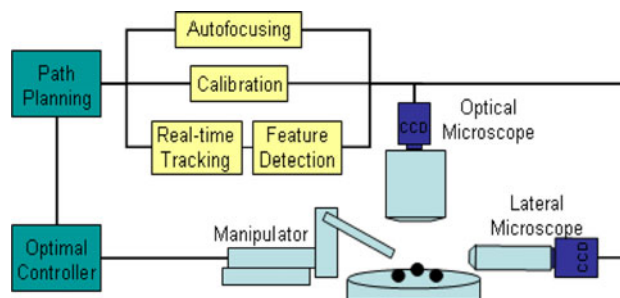


Fig. 1 System diagram with all vision modules and hardware–software components

is successfully demonstrated to show the potential of our developed algorithms.

In dealing with the microworld, there are lots of uncertainties associated with the optical system and the objects to be manipulated. For example, to have a coarse view of the workspace where probes and objects are located, one may need relatively small magnifications, whereas in manipulations tasks one will typically need larger magnifications for fine view. So, what is really needed is a system that allows manipulation under different experimental conditions such as varying lighting, different working distance and variable zooming. Direct utilization of well-known vision algorithms cannot handle these situations. For example, detecting the probe tip is not a straightforward point feature detection problem. The probe tip must be extracted in a robust manner during the whole manipulation scenario, which is not a trivial issue. Similarly, direct application of Tsai's well-known calibration algorithm does not work because the microscope's objective plane is near parallel to the calibration pattern located on the sample stage, which violates conditions to be satisfied by Tsai's algorithm. Therefore, certain modifications on Tsai's algorithm must be introduced. Based on these observations, one should consider vision modules that are, to a certain extent, robust to various uncertainties associated with the optical system and the manipulated objects. The vision modules developed in this paper provide relatively robust solutions to micromanipulation tasks and, therefore, are termed as robust vision modules.

However, we do not claim that our proposed vision modules can work under all possible experimental conditions. For example, in manipulating microspheres backlighting is used to illuminate the sample stage, since the illumination from above makes the image processing harder and unstable due to the reflectance from the shiny surface of the polystyrene balls. So, if direct illumination from above is used, object detection module may have trouble in dealing with shiny surfaces of the microballs. Besides, the real-time tracking algorithms implemented in our paper can process images at 30 fps. Some micromanipulation tasks may require faster tracking algorithms.

The organization of this text is as follows: Sect. 2 introduces robust vision modules. Section 3 is on visual controller synthesis. Section 4 develops a visual path-planning algorithm. Experimental results obtained with a microassembly workstation and discussions of the results are presented in Sect. 5. Finally Sect. 6 concludes the study with some remarks and gives an outlook for future works.

2 Robust vision modules for microsystems

2.1 Autofocusing

It is challenging to use visual feedback in micromanipulation due to the small depth of field in microscopes. When we use microobjects with different heights, inclined surfaces or cameras with optical axis not perpendicular to the scene, only small portions of the workspace might be in focus. However, object detection algorithms often demand sharply focused images to extract useful information. Thus, it is important to capture focused images of a moving image feature such as the tip of a probe in the 3D space during a micromanipulation task. To use the visual feedback efficiently in the experiments, we developed an autofocus module to continuously have sharp images of the features tracked.

According to Krotkov [18], the first problem of automatic focusing is: given the projection $P' = (u, v)$ onto the focal plane of an object point $P = (x, y, z)$ (z unknown), what focal length produces the sharpest definition of P' ? The second problem is: given the focal length acquired from autofocus, to recover the z component of P , what is the depth? Our main consideration in this study is the first one.

In the literature, a variety of algorithms to measure the sharpness of an image are proposed and compared [18, 19]. These methods can be classified into three main groups: derivative, statistics and histogram-based methods. The derivative based methods relate the focus measure with the high frequency content in the images. The second group searches for focused images using correlation and/or variance. The third focus measure is determined by exploiting the histogram of spatial and frequency intensities in the image. To judge the performance of the focus measures for the manipulation experiments, the best three criteria, [tenengrad (T), energy laplace (EL), and normalized variance (NV)] are evaluated. While the first two methods are derivative based, the third one is a statistics based method. The focus criterion functions for the measures are given as:

$$F_T = \sum_W \sum_H I_x^2(u, v) + I_y^2(u, v) \quad (1)$$

$$F_{EL} = \sum_W \sum_H C(u, v) \quad (2)$$

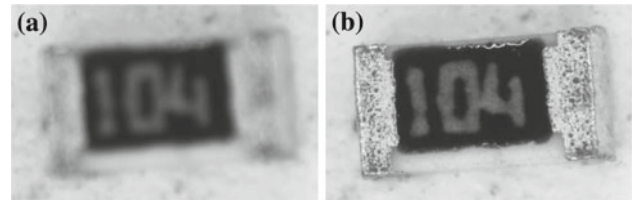


Fig. 2 **a** Defocused and **b** focused images

$$F_{NV} = \frac{1}{H \times W \times \mu} \sum_W \sum_H (I(u, v) - \mu)^2 \quad (3)$$

where W is the width, H is the height, and I_x and I_y are the derivatives of the image I along x and y directions, respectively. $C(u, v)$ is the second derivative of I image and μ is the mean of the intensities in I .

To test the algorithms, a sequence of the images at different Z coordinates are captured. Since all the regions in the field of view do not have the same height, a planar patch on the captured object using 50×50 pixels window are chosen to be processed during the experiment. The microscope moves along the z axis in a range of $1,750 \mu\text{m}$ with $5 \mu\text{m}$ steps. To eliminate the displacement of the object in x and y directions due to the tilt angles between the camera and the sample frame during the z movement, the center of the monitored object is aligned with the image center. The positioning stages are driven to minimize the distance between the image and object center using the visual feedback. Figure 2 shows two aligned images of a resistance with the dimensions $2 \times 1 \times 0.5 \text{ mm}$ from the captured array at different depths.

The results of the presented focus measures are depicted in Fig. 3. Note that to make valid comparisons, the focus curves are normalized.

Before evaluating the results, it should be known that creating an image sequence with small linear incremental steps is a very slow process. Thus, the optimal focus measure should be unimodal, monotonic, and should reach the maximum only when the image is focused to support a faster algorithm [20]. According to this criterion, the statistics-based algorithm (normalized variance) outperforms the other two algorithms. It is more selective to the focus changes and has less local maxima. Although this algorithm amplifies the effect of noise, the images can be pre-processed before computing the focus measure.

Based on the unimodal assumption of the focus measure profile, Fibonacci search is used for finding the sub-interval which includes the maximum of the normalized variance focus measure, $F_{NV}(z)$, for a given interval $[a, b]$. The idea is to move the microscope along the z axis starting from its home position ($z = a$), at which microobjects can be roughly seen, and search for the sub-interval which contains the maximal focus measure computed from the image.

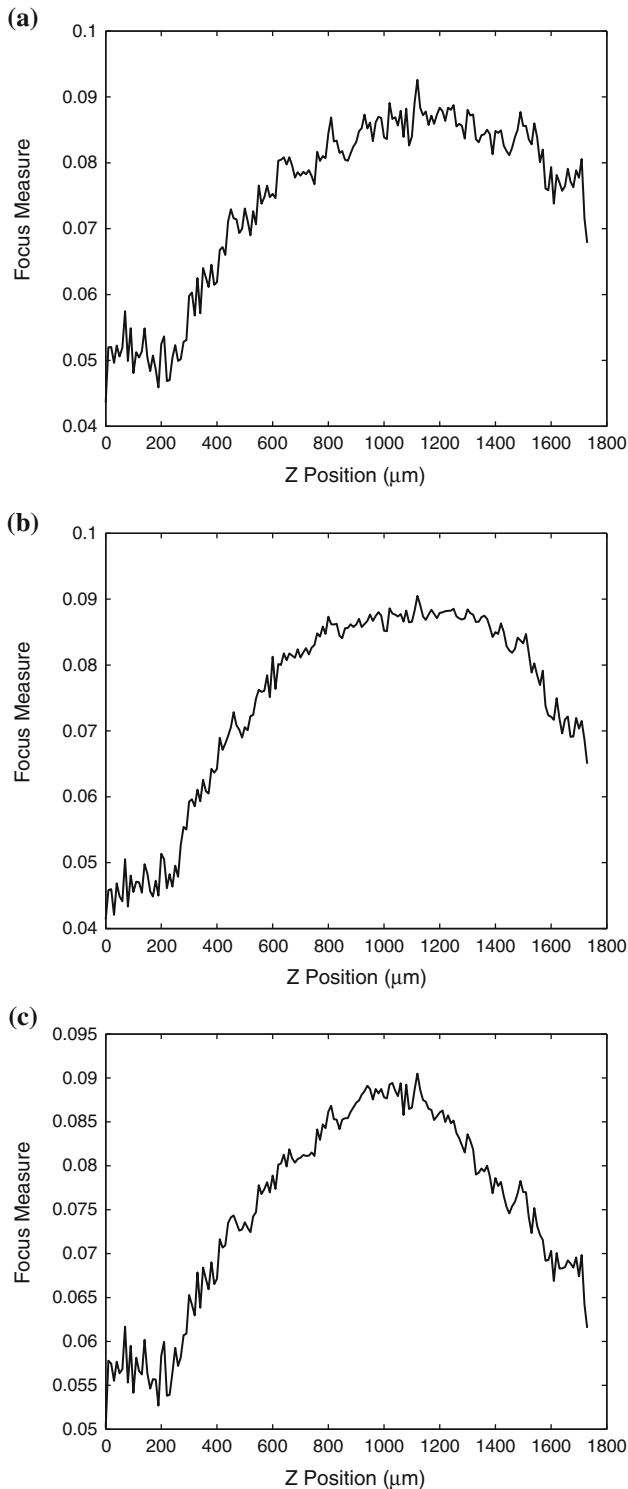


Fig. 3 Results for **a** Tenengrad, **b** Energy Laplace, **c** normalized variance

It should be noted that the computed sub-interval is limited with the motion resolution of the motor, which moves the microscope along the z axis in the experiments.

To be more precise, let z_m denote the m th element in $[a, b]$ and let $F_{NV}(z_m)$ denote the computed focus measure at that element. Although the elements of the interval are sorted either in descending (i.e. $z_{m-1} > z_m > z_{m+1}$) or ascending (i.e. $z_{m-1} < z_m < z_{m+1}$) order for every $m < N \in \mathbb{Z}^+$ where N is the number of elements in the input array, $F_{NV}(z_m)$ values need not be sorted, otherwise the first or the last term in the array would be found in the first place. Note that computing all the focus values on the list is a time-consuming process and Fibonacci search is employed to avoid this problem.

The function $F_{NV}(z)$ is unimodal on the interval $[a, b]$, if there exists a single point $p \in [a, b]$ such that $F_{NV}(z)$ is increasing on $[a, p]$ and decreasing on $[p, b]$ (for the maximum search) or vice versa (for the minimum search).

Let K_i denote the i th Fibonacci number where

$$K_0 = 0, K_1 = 1, K_i = K_{i-1} + K_{i-2} \quad \text{for } i = 2, 3, \dots$$

Assuming $b > a$, a pseudo-code of Fibonacci search is given in Algorithm 1.

Algorithm 1 Fibonacci Search

Fibonacci Search (a, b, ϵ)

```

{
     $a_0 = a$ 
     $b_0 = b$ 
     $K_0 = 0$ 
     $K_1 = 1$ 
     $i = 2$ 

    while
    {
         $K_i = K_{i-1} + K_{i-2}$ 
         $a_i = a_0 + (1 - K_{i-1}/K_i) * (b_0 - a_0)$ 
         $b_i = a_0 + (K_{i-1}/K_i) * (b_0 - a_0)$ 

        if ( $F_{NV}(a_i) > F_{NV}(b_i)$ )
        {
             $a_0 = a_i$ 
             $b_0 = b_i$ 
        }
        else
        {
             $a_0 = a_i$ 
             $b_0 = b_i$ 
        }
        if ( $b_0 - a_0 < \epsilon$ )
        {
            return  $[a_0, b_0]$ 
        }
         $i = i + 1$ 
    }
}

```

Fibonacci search may be considered as an improved version of the binary search. Instead of using the middle element at each step, we guess more precisely where the element being sought falls within the current interval of

interest. Instead of dividing the array in the middle, Fibonacci search splits the array based on the Fibonacci numbers. Both Fibonacci and binary search have an average complexity of $O(\log(n))$.

At a_i and b_i , the sharpness is calculated according to the normalized variance focus measure. If sharpness at a_i is bigger, the new interval is $[a_0, b_i]$, otherwise it is $[a_i, b_0]$. The search algorithm proceeds that way and at each step, the interval where the highest sharpness will be searched becomes smaller. As a result, the algorithm finds at which depth the highest sharpness occurs and acquires the sharpest image.

It should be noted that different length of intervals are given as input to the search algorithm for different magnification levels. It is also ensured that the given interval always includes the best-focused z coordinates before Fibonacci search is started.

Defocused and focused images of two samples using the presented algorithm are depicted in Fig. 4.

2.2 Object and probe tip detection

Manipulating microobjects with the probe requires the knowledge of position of the probe and the object. To obtain these information, we might need a priori information about the size and shape of the probe and the object to be manipulated. In the micromanipulation experiments, micro polystyrene balls with different diameters (between 8 and 70 μm) are manipulated by the probe, which is mounted on the x , y , z positioning stage.

Having characterized the geometrical properties of the objects and the probe, it is important to decide what information has to be extracted from the image. The center of gravity of an object, corners, area of a surface and distance between two points can be considered as image features.

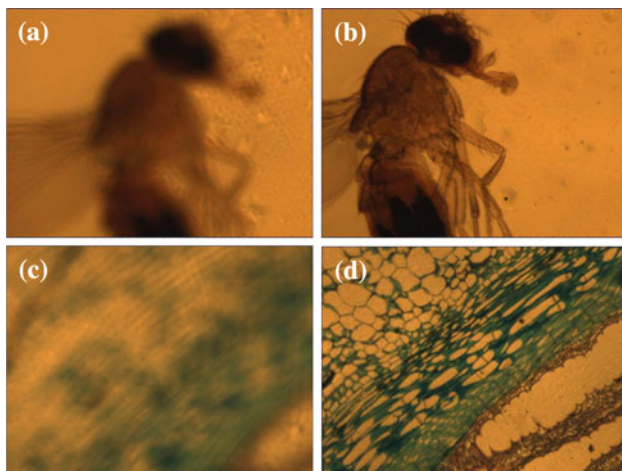


Fig. 4 **a** Defocused and **b** focused images of a *Drosophila*. **c** Defocused and **d** focused images of pumpkin cells

In our micromanipulation experiments, it is aimed to push the microobjects to the desired locations accurately using a probe. To facilitate such a scenario, we need to have the information about the center of gravity of the microball, its diameter, the tip location and orientation of the probe. Thus, several image-processing algorithms, using a priori size and shape information, are developed to obtain the desired features in the setup.

Since the probe moves in the three-dimensional space, it is vital to detect the three-dimensional coordinates of the contact point on the ball. Thus, real-time extraction of that point from the top and side cameras becomes crucial. Since the top and side views can be assumed to be calibrated, the same algorithm is implemented for the images from the cameras. Since the cameras have different configurations and have been exposed to different illumination effects, different threshold values are assigned for the two views.

The first step in the circle detection algorithm aims to remove the noise and the perturbations by using a smoothing filter. An illustrative top view of the work cell, which contains several microballs and some dust particles, is shown in Fig. 5.

One can observe that backlighting is used to illuminate the sample stage, since the illumination from above makes the image processing harder and unstable due to the reflectance from the shiny surface of the polystyrene balls.

The next step is to differentiate the balls from the background. Since the backlight illumination power is controllable, it is experimentally observed that a large number of pixels are gathered around the intensity value of 235, which is the mean background grayscale value (See Fig. 6a). A value of 215, which is approximately 90% of the mean background grayscale value, is then set as a threshold. The thresholded image is depicted in Fig. 6b.

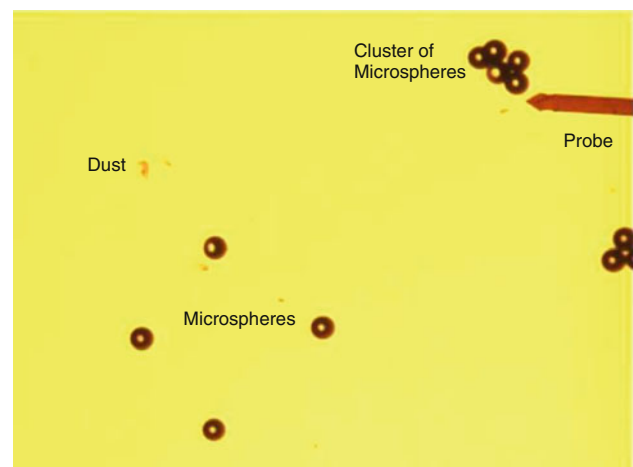


Fig. 5 Top view of the sample stage

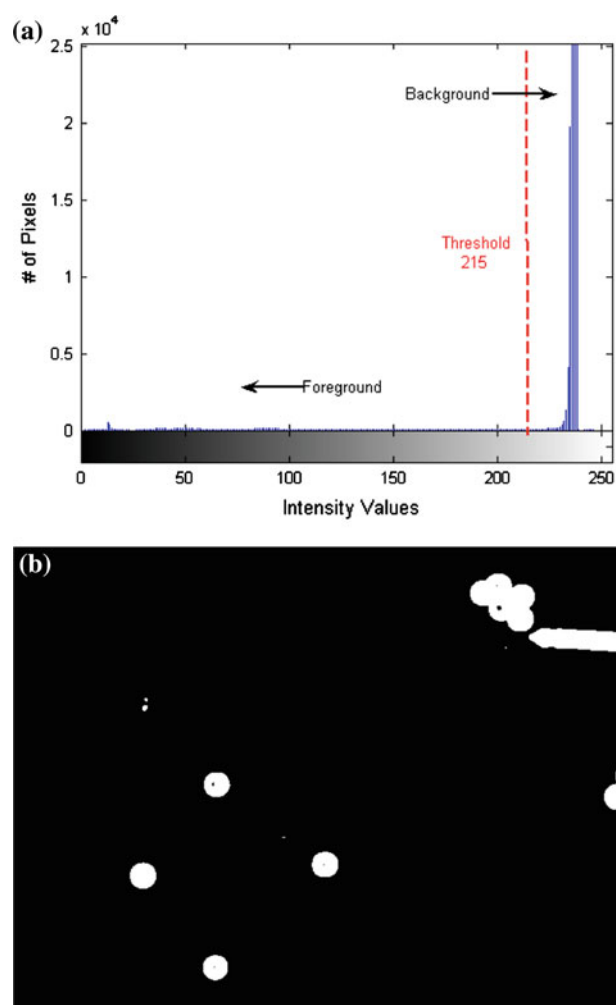


Fig. 6 **a** Histogram of the sample stage image, **b** image after thresholding

We should note that this is not a global threshold that can be used for every microscope/camera setting and/or experimental working condition. Since almost the same amount of light was used and experiments were performed under a few magnification conditions, we selected the above threshold value based on the histogram of image pixels. When working with polystyrene balls using backlighting, this threshold value worked quite well. However, we cannot conclude that it will work under all lighting and magnification conditions.

In the next step, a connected component finder algorithm is employed to retrieve all the contours and organizes them into two-level hierarchy for the thresholded image in the previous step. As the connected components are found, the contours with areas in the predetermined range are chosen as the region of interests. Now, we can eliminate the regions where the clusters of microballs and relatively bigger dust from the microspheres stand alone. Note that the cluster of balls are not chosen to be manipulated due to difficulty in separation.



Fig. 7 Segmentation result

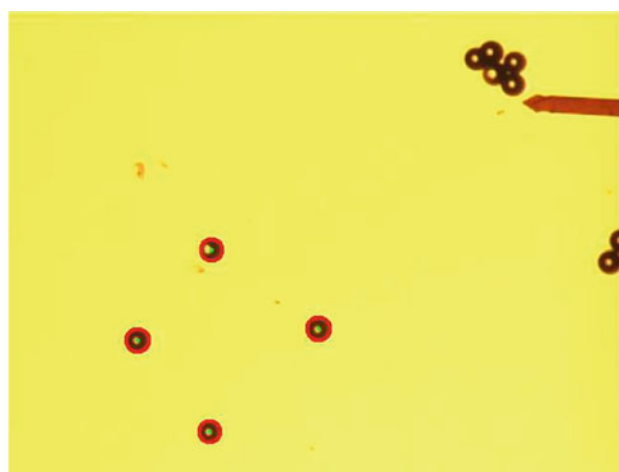


Fig. 8 Extracted circles

The resulting segmentation yields the regions, which contain only the individual microspheres, as depicted in Fig. 7.

Having the balls segmented, the generalized Hough transform, which is robust under noise is implemented to detect the spheres, as shown in Fig. 8. It should be noted that the Hough transform with a minimum distance criteria between the centers might be employed to detect the circles in the entire frame without using the previous steps. However, it would not allow a real-time performance. On the other hand, using the presented algorithm, the center positions and radii of the spheres are detected in real-time.

It is vital to detect and update the position of the probe from the top and side cameras during the experiments to increase manipulation precision. Since the probe is moved in the 3D, two individual algorithms are required to detect the position of the probe in the x - z and x - y plane.

The geometries of the end effector in the x - y and x - z planes, which can be measured with the two microscopes (optical and close focus), are constant. It is well known that

template matching methods give good results for this sort of detection problems. Template matching compares for the best match with a reference template by sliding the template from the top left to the bottom right of the image. One of the drawbacks of template matching is its large computational cost. The size of the template and image are the factors that determine the computation time. However, the template size is also important for the robustness of detection. Thus, an optimal size of image with a region of interest and template is chosen in the experiments. Another shortcoming is that template matching does not give subpixel results, unless it is modified with an interpolation algorithm. Since it is not required to have subpixel accuracy in the x - z plane (side camera), the probe is detected by using template matching with the normalized cross correlation (NCC) technique [21],

$$NCC = \frac{\sum_{x,y} (I(u+x, v+y) - \bar{I}_{u,v}) T(x, y)}{\sqrt{\sum_{x,y} (I(u+x, v+y) - \bar{I}_{u,v})^2 \sum_{x,y} T(x, y)^2}} \quad (4)$$

where $I(u, v)$, $T(x, y)$ and $\bar{I}(u, v)$ denote the image, the template and the local image mean at location (u, v) respectively. The brightness of the image and template can vary due to lighting conditions during the experiments. Thus, NCC method that uses the normalized images is employed to eliminate the illumination effects.

Computing the contact point of the probe in the x - y plane is vital to locate microspheres to the desired targets precisely. Therefore, the detection of probe tip is computed in subpixel accuracy by exploiting the known geometry of the probe. In the algorithm, the edge points of the probe are extracted by the Canny edge detector. The original image and edge map of the probe are shown in Fig. 9.

Since we have a priori information about the geometry and the orientation of the probe, the edge pixels with the predetermined slopes can be fitted to two parallel lines in a subpixel accuracy. It is assumed that the probe tip is on the line that is parallel and in the middle of the previously extracted lines. The exact coordinate of the tip is computed through searching the first edge pixel along the line. The detection algorithm is illustrated in Fig. 10.

Note that in the presented detection algorithms in the x - y and x - z planes, it is assumed that the home position of the probe is known.

2.3 Real-time tracking

After the initial detection of the object and the probe, it is not required to process the entire image for the following frames. Having computed the required image features of object (x_{k-1}^{obj}) and the probe tip (x_{k-1}^{tip}) for $t = k - 1$, the positions can be estimated using an estimator for the next

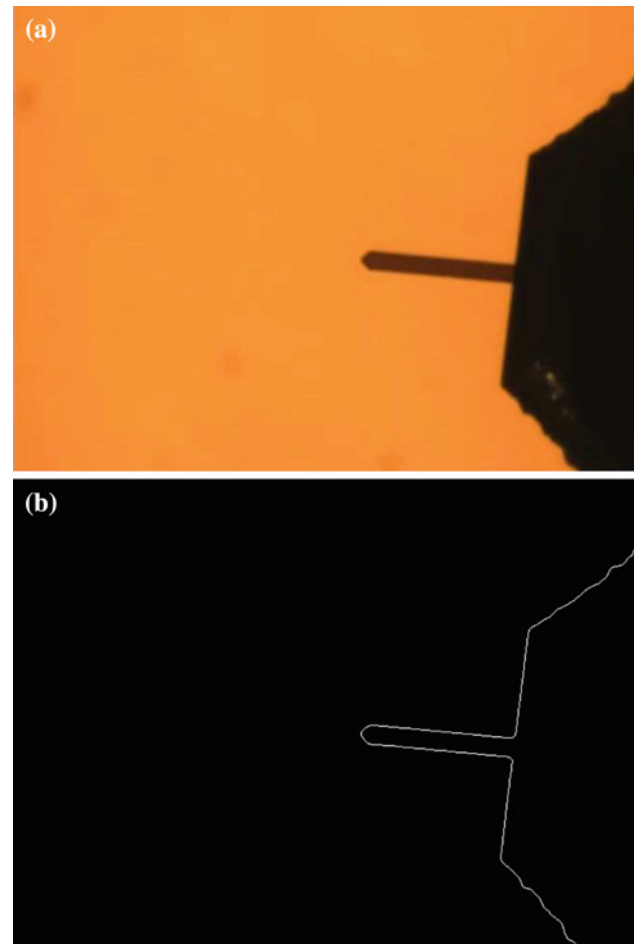


Fig. 9 **a** Original image of the probe. **b** Extracted edge pixels of the probe

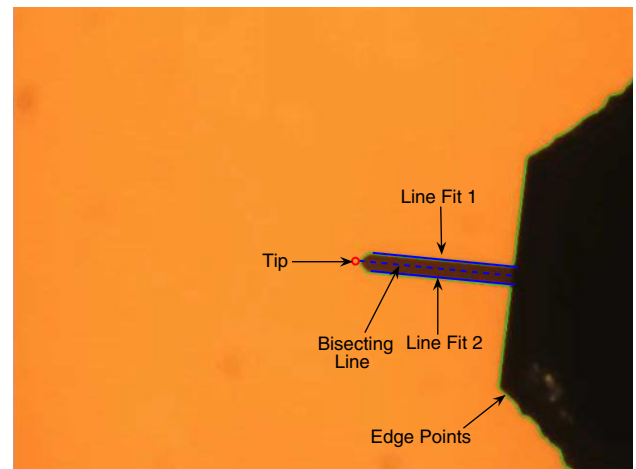


Fig. 10 Computed image features for the probe

time step $t = k$. Assuming that the motions of the object and the probe are governed by the linear difference equations,

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad (5)$$

with a measurement $z \in \mathbb{R}^2$

$$z_k = Hx_k + v_k \quad (6)$$

where $x_k \in \mathbb{R}^4$ is the state vector which contains (x, y) position and velocities of the object (or the probe), and $A \in \mathbb{R}^{4 \times 4}$ is the state matrix given by

$$A = \begin{pmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where T is the visual sampling time between two consecutive frames. The matrix $B \in \mathbb{R}^{4 \times l}$ relates the optional control input $u \in \mathbb{R}^l$ to the state x_k . Unlike a control problem, there is no explicit control input in a visual tracking problem, and we therefore set $u_{k-1} = 0$. Thus, the B matrix is not employed in the tracking problem. The matrix $H \in \mathbb{R}^{2 \times 4}$ relates the state x_k to the measurement z_k and is defined as

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

The random variables w_{k-1} and v_k represent the white process and measurement noise with normal probability distributions, respectively.

Based on the model in (5) and (6), the estimated object center \hat{x}_k^{obj} and the probe tip \hat{x}_k^{tip} can be computed using the discrete Kalman filter [22]. Once we have \hat{x}_k^{obj} and \hat{x}_k^{tip} , these coordinates can be used to define new regions of interest at $t = k$, in which the detection is performed. Employing a region of interest improves the computation time and robustness of the mentioned detection algorithms. As a result, the locations of two image features, position of the probe tip and the sphere which is being pushed are tracked by the tracking module in real-time during the micromanipulation experiments.

We should remark that tracking of the object and the probe are not performed simultaneously on the x - y and x - z planes. Tracking is done sequentially. In other words, first motion of the probe is tracked on the x - z plane until it comes in contact with the objects. Its motion is then tracked on the x - y plane.

To perform tracking simultaneously on both the x - y and x - z planes, projection equations of both microscopes (the top and the side microscopes) must be optimized in the same cost function during calibration. In fact, this was tried but did not work well since the image quality obtained from the side microscope was not good.

Accurate positioning and trajectory tracking results obtained by visual servoing experiments presented in Figs. 17–21 and the successful micromanipulation tasks presented in Figs. 11, 22 and 23 show the effectiveness of the real-time tracking algorithm. We should also remark that in a recent study [23] where we introduced a novel online calibration algorithm, we used the same real-time tracking

algorithm based on Kalman filter and obtained 0.1 pixel error. This level of accuracy cannot be reached by using a poor tracking algorithm.

2.4 Optical system calibration

Visual information is the crucial feedback to enable the micromanipulation and microassembly tasks. Processing visual data determines the path of the probe in the image frame; however, the input to the manipulator is given in its own frame. Thus the mapping between the manipulator frame and the image frame forms a critical component for servoing the probe and the gripper. To compute the mapping, a calibration method is developed and implemented.

Several calibration methods exist in the literature that are mostly used in computer vision applications [24, 25]. However, these methods cannot directly be employed to calibrate an optical microscope coupled with a CCD camera due to the unique characteristics of the optical system. Large numerical apertures, high optical magnifications and very small depth-of-field property of optical microscopes restrict the calibration to a single parallel plane. Although some methods [26, 27] were proposed to calibrate the optical microscope, they are computationally complex and cannot propose a solution for the close focus microscope, side view, since it does not have the same image forming components with a typical microscope. Therefore, a simple modification to Tsai's algorithm [24] is done via weak perspective camera model assumption.

The steps of the calibration algorithm are as follows:

1. The probe is detected using a template-matching method and is started to be tracked.
2. The controller generates certain number of points for the probe, which corresponds to the corners of a virtual calibration grid.
3. The pixel coordinate of the probe is computed using the normalized cross correlation (NCC) technique for each given position.
4. Once the positions of the probe in camera and manipulator space are recorded, the radial alignment constraint (RAC) [24] is employed to compute the rotation and translation from the manipulator coordinate frame to the image coordinate frame.
5. The total magnification (M) of the system and the radial distortion coefficient (κ_1) can be obtained by a least square solution, which minimizes the following error:

$$E = \frac{\sum^N [(x_i - \tilde{x}_i)^2 + (y_i - \tilde{y}_i)^2]^{1/2}}{N} \quad (7)$$

where E is the error, N is the total number of points, (x_i, y_i) are the pixel coordinates of an image point and $(\tilde{x}_i, \tilde{y}_i)$ are

coordinates of the projected world point for the computed rotation and translation.

The first three steps in the calibration algorithm given above are based on the detection and tracking of the probe, which were explained in detail before. We will explain the last two steps [Steps (4) and (5)] in this algorithm in more detail below.

In the optical system calibration, three coordinate frames, namely F^o : objective, F^i : image and F^w : world frames are used. The origin of the objective coordinate system is the optical center (o), and the X^o and Y^o axes are aligned with the rows and columns of image frame, respectively. The Z^o axis is aligned with the optical axis of the microscope. The origin of the image coordinate system is the intersection of the virtual image plane with the optical axis and X^i , Y^i are parallel to the X^o and Y^o axes. The world frame can be chosen arbitrarily. However, it is more convenient to attach the world frame to the tip of the probe.

The transformation from the world frame to the objective frame is given by a rotation matrix (R) and a translation vector (T)

$$\begin{pmatrix} X^o \\ Y^o \\ Z^o \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} X^w \\ Y^w \\ Z^w \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \quad (8)$$

We can write the image of a point (X^o , Y^o , Z^o) in the undistorted image coordinates (x' , y') as

$$x' = \frac{f}{s_x} \frac{X^o}{Z^o} + o_x \quad (9)$$

$$y' = \frac{f}{s_y} \frac{Y^o}{Z^o} + o_y$$

where f is the objective focal length, s_x and s_y are horizontal and vertical pixel sizes, respectively, (o_x , o_y) are the coordinates of image center. Combining Eq. (8) with (9) gives

$$x' - o_x = f_x \frac{r_{11}X^w + r_{12}Y^w + r_{13}Z^w + T_x}{r_{31}X^w + r_{32}Y^w + r_{33}Z^w + T_z} \quad (10)$$

$$y' - o_y = f_y \frac{r_{21}X^w + r_{22}Y^w + r_{23}Z^w + T_y}{r_{31}X^w + r_{32}Y^w + r_{33}Z^w + T_z}$$

where $f_x \doteq f/s_x$ and $f_y \doteq f/s_y$.

When working with optical microscopes one should realize that the object plane, i.e. sample stage, under the microscope is nearly parallel with the image plane. This implies that $r_{13} = r_{23} = r_{31} = r_{32} \simeq 0$ and $r_{33} \simeq 1$. Note that the calibration object on the sample stage cannot be tilted arbitrarily due to the small depth of field associated with the optical microscope. This is in contrast with the calibration conditions dictated by Tsai's algorithm where one should consider tilting the calibration pattern up to several degrees.

Since the depth of the object itself is much smaller than the average distance (\bar{Z}) along the optical axis, the undistorted image coordinates of an object can be written in the objective

frame as

$$x' \approx f_x \frac{X^o}{\bar{Z}} = M_x X^o \quad (11)$$

$$y' \approx f_y \frac{Y^o}{\bar{Z}} = M_y Y^o$$

where $M_x \doteq f_x/\bar{Z}$ and $M_y \doteq f_y/\bar{Z}$ are the magnifications along the x and y axes of the objective. We assume that the coordinates of the image center is known. Thus, without loss of generality, we can consider translated image coordinates, $(x', y') = (x - o_x, y - o_y)$. It is also assumed that only the first term of the radial distortion is dominant. Thus,

$$\begin{aligned} x' &= x(1 + \kappa_1 r^2) \\ y' &= y(1 + \kappa_1 r^2) \end{aligned} \quad (12)$$

where (x, y) are the distorted image coordinates of a point, κ_1 is the radial distortion coefficient and $r = \sqrt{x^2 + y^2}$ is the radial distance from the distortion center. Combining the above equations gives the following relation between the image coordinates and world coordinates in terms of the parameters to be calibrated (assuming $Z^w = 0$)

$$\begin{aligned} x(1 + \kappa_1 r^2) &= M_x(r_{11}X^w + r_{12}Y^w + T_x) \\ y(1 + \kappa_1 r^2) &= M_y(r_{21}X^w + r_{22}Y^w + T_y) \end{aligned} \quad (13)$$

We now solve for the unknown parameters. First observe that the two equations in (13) can be divided to eliminate $1 + \kappa_1 r^2$. This is the RAC (radial alignment constraint) in Tsai's algorithm [24] and implies the following equation

$$x(r_{21}X^w + r_{22}Y^w + T_y) = ay(r_{11}X^w + r_{12}Y^w + T_x) \quad (14)$$

where $a \doteq M_x/M_y = f_x/f_y = s_y/s_x$ is the aspect ratio of the pixels. Equation (14) can be rewritten as

$$\begin{pmatrix} yX^w & yY^w & y - xX^w & -xY^w \end{pmatrix} \begin{pmatrix} aT_y^{-1}r_{11} \\ aT_y^{-1}r_{12} \\ aT_y^{-1}T_x \\ T_y^{-1}r_{21} \\ T_y^{-1}r_{22} \end{pmatrix} = x \quad (15)$$

With N (the number of object points) much larger than 5, an overdetermined system of linear equations can be formed and solved for the five unknowns, $aT_y^{-1}r_{11}$, $aT_y^{-1}r_{12}$, $aT_y^{-1}T_x$, $T_y^{-1}r_{21}$, and $T_y^{-1}r_{22}$. Once these unknowns are determined, one can extract a , T_y , T_x and the elements of the rotation matrix r_{ij} from these unknowns using the orthonormality of the rotation matrix. The reader can consult [24] for the details of this procedure. Hence, in the following development, a , R and the first two components T_x and T_y of T are assumed to be known.

Minimization of the error E in Eq. (7) in terms of f_x , T_z and κ_1 , under the assumption of perspective projection given by Eq. (10), is a nonlinear problem, and can be solved through gradient descent techniques. However, since we have made

a weak-perspective assumption in Eq (11), the third component of the translation T_z is absorbed in \bar{Z} , which is in turn absorbed in $M_x = f_x/\bar{Z}$ and $M_y = f_y/\bar{Z}$. Thus, estimation of κ_1 and M_x becomes a linear problem as we show below. First, we make the following substitutions.

$$\begin{aligned} m &= r_{11}X^w + r_{12}Y^w + T_x \\ n &= r_{21}X^w + r_{22}Y^w + T_y \end{aligned} \quad (16)$$

We then add the two equations in (13) and utilize Eq. (16) to get

$$M_x(m + n/a) - \kappa_1(x + y)r^2 - (x + y) = 0 \quad (17)$$

which is linear in κ_1 and M_x and can be rewritten as

$$(-(x + y)r^2 \quad m + n/a) \begin{pmatrix} \kappa_1 \\ M \end{pmatrix} = x + y \quad (18)$$

where κ_1 and M are the unknowns to be determined. With N (the number of object points) much larger than 2, an over-determined system of linear equations can be formed and solved for the two unknowns, κ_1 and M_x .

These linear estimates can always be refined through nonlinear optimization using gradient descent techniques. In such a case, these estimates will be used as initial guess for the nonlinear optimization algorithm.

The average calibration error ranges from 0.01 to 0.08 pixel for the coarse, fine and side views. Since the magnifications of the lenses and cell sizes of cameras can differ, the average error in metric coordinates varies from ten nanometers to less than a micron.

3 Visual servoing

The complex geometry of the observed biological samples and high numerical apertures of optical microscope, which results in small depth of field, lead to a challenging 3D pose estimation problem. Therefore, an image-based visual servoing approach is preferred in the micromanipulation tasks. Image-based visual servoing methods typically use image features such as points, lines, moments, etc. For a recent survey of visual servo control methods and some other works on vision-based control strategies, the reader may look at [28–35]. In addition, eye-to-hand configuration is assumed in the vision-based control design, since the vision sensors are stationary in the setup.

The key equation which relates the temporal derivative of 2D visual features to the six dimensional velocity screw (3 translational and 3 rotational velocities) of the robot end effector is given as follows:

$$\dot{s} = L_s \mathbf{V} \quad (19)$$

where $s \in \mathbb{R}^n$ is the vector of image features being tracked, n is the number of the features, $L_s \in \mathbb{R}^{n \times 6}$ is the so-called “interaction matrix” or “image jacobian”, and \mathbf{V} is the velocity screw of the end effector.

To synthesize a vision-based closed-loop controller, we can first define the feedback error, which is the difference between the actual (s) and the desired (s^*) visual features, namely

$$e = s - s^* \quad (20)$$

By taking time derivative of Eq. (20) and using Eq. (19), it follows that

$$\dot{e} = \dot{s} - \dot{s}^* \Rightarrow \dot{e} = L_s \mathbf{V} \quad (21)$$

where it is assumed that $\dot{s}^* = 0$, i.e. the target is stationary.

Thus, considering \mathbf{V} as the velocity input (translational and rotational velocities) to the robot controller, a simple proportional controller, which ensures an exponential decrease of the error ($\dot{e} = -\lambda e$), can be designed as

$$\mathbf{V} = -\lambda L_s^\dagger e \quad (22)$$

where λ is a positive controller gain and $L_s^\dagger \in \mathbb{R}^{6 \times n}$ is the pseudo inverse of L_s . Derivation of the interaction matrix for a point feature can be found in [28]. Entries of L_s are functions of the intrinsic parameters of the optical system and the depth. Intrinsic parameters are obtained through the optical system calibration module detailed in Sect. 2.4.

It is also possible to design control laws that optimize various system performance measures. To this end, the relationship given in equation (19) can be written in discrete time as

$$s(k+1) = s(k) + T L_s(k) \mathbf{V}(k) \quad (23)$$

where T is the sampling time of the vision sensor.

Employing the optimal control techniques in [34], the cost function can be defined to penalize the pixelized position errors and the control energy as

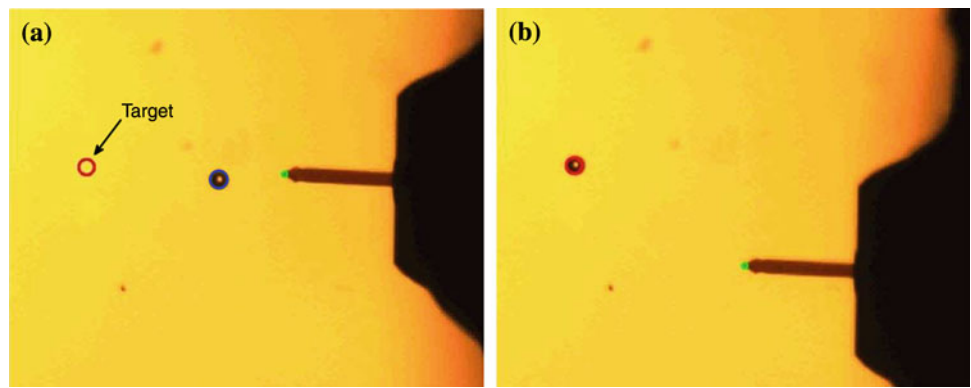
$$\begin{aligned} E(k+1) &= (s(k+1) - s^*(k+1))^T Q (s(k+1) - s^*(k+1)) \\ &\quad + \mathbf{V}^T(k) R \mathbf{V}(k) \end{aligned} \quad (24)$$

In light of Eq. (23), the resulting optimal control input $\mathbf{V}(k)$, which minimizes the above cost function, can be derived as

$$\mathbf{V} = -(T L_s^T Q T L_s + R)^{-1} T L_s^T Q (s - s^*) \quad (25)$$

where k is dropped for notational simplicity. The weighting matrices Q and R can be adjusted to ensure the desired response.

Fig. 11 Automatic micromanipulation of a microsphere, **a** initial position, **b** final position



4 Path planning

In the previous sections, we developed necessary vision and control modules to manipulate the microparticles. Having the object detection, tracking, calibration and visual servoing modules, we are now able to propose an algorithm to push a microsphere along a designed path and locate it to the desired target position. An illustrative scene for the problem task is given in Fig. 11.

The stated micromanipulation task for a microsphere can be extended to the scenarios, which aim to move several microparticles serially along the computed path and to generate different patterns.

The algorithm to achieve the desired pattern formation is given below:

1. The side camera determines the optimal z position by processing the sample stage scene and the probe is moved to the determined contact point in the x - z plane.
2. Workspace is explored, the particle and obstacle map are recorded before the workspace is occluded by the moving probe.
3. If there are not adequate microballs for the predetermined pattern, the program is terminated.
4. The closest microsphere to the target is chosen as the first particle to be pushed so that there will be no obstacle along the pushing line.
5. Unless there is no obstacle along the line between the probe tip and center of the particle, the probe directly approaches the destination at a given speed in the x - y plane. If any obstacle is detected, the obstacle avoidance determines the path until no obstacle exists along the way to the particle.
6. The probe is moved to the determined contact point in the x - y plane by having visual feedback from the top camera. (the manipulator x - y plane, this step is done only once).
7. The particle is pushed along the line that connects the center of the sphere to the target until the particle is located at the target.
8. After the recorded obstacle map (in the second step) is reprocessed, the probe is moved to its initial position directly, if there is no obstacle along the path to the initial position. If any obstacle is detected on the way, obstacle avoidance mode is activated until no obstacle exists along the line between the probe tip and its initial position.
9. Go to Step 2 until the predetermined pattern is generated.

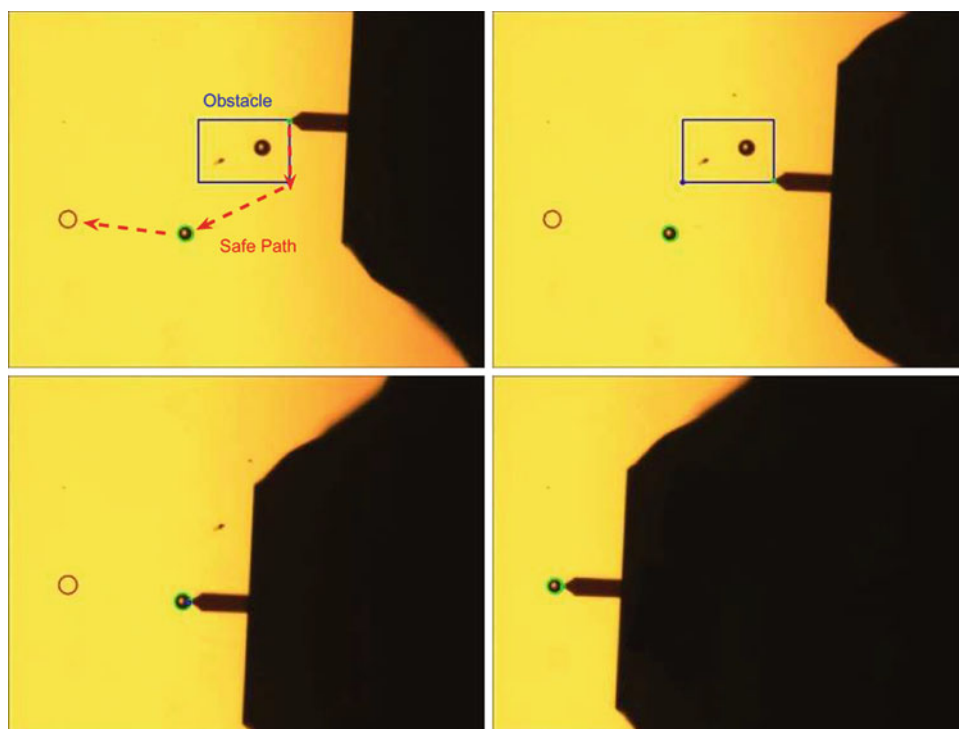
The serial micromanipulation process includes five stages such as z movement, explore, obstacle avoidance, pushing and home modes. In the z movement mode, the probe tip is positioned along the z axes by processing the workspace image acquired from the side camera. In the explore mode, the probe is positioned such that all the workspace is visible and no occlusion occurs due to the probe position. Thus the global particle map can be extracted and used for the following steps. As the probe approaches the determined particle, there may be an obstacle on the line to the destination. Then the region twice the size of the obstacle is surrounded by a rectangular path and it is followed by the probe until no obstacle exists on the way to the particle. An illustrative figure is shown in Fig. 12.

In the home mode, the probe is moved to its initial position with the obstacle avoidance support so that the particle map in the workspace can be updated with no occlusion of the probe.

5 Experimental results and discussion

5.1 Microassembly workstation

In the microworld, there exist uncertainties about the position, behavior and shape of the objects to be manipulated. Parts may stick, flip away and deformed with an applied force in micromanipulation tasks. In addition, the shapes of micron

Fig. 12 Obstacle avoidance

sized objects may not agree with the designed geometry, since manufacturing systems cannot ensure very high precisions in μm . Therefore, a microassembly workstation was developed to compensate uncertainties and to manipulate small parts with small tolerances at high accuracies. The model of the overall experimental system setup is shown in Fig. 13.

The aforementioned functional requirements for the microassembly workstation call for the use of external sensors to guide manipulation tasks. Thus, three types of feedback are provided from the vision, position and force sensors in the system. The vision system constitutes the key component in the setup, since high precision micromanipulation is strongly dependent on the visual feedback which

allows direct measurement of positions and orientations of the objects and the probe. A visually guided manipulation implies the use of monitoring devices with high magnifications such as microscopes. Based on the specifications of a micromanipulation scenario, the size of the object to be manipulated and thus the required resolution and field of view may vary.

In the microassembly workstation, different magnification and resolution levels are available. In order to allow global and local visual information, a coarse and a fine view with variable zooming are employed. These cameras are mounted on a stereo optical microscope, Nikon SMZ1500 with $1.5\times$ objective and $0.75:11.25$ zoom ratio. While an A602fc Basler with $9.9 \times 9.9 \mu\text{m}$ cell sizes provides coarse view for the sample stage, a Sony XCD-X710CR with $4.7 \times 4.7 \mu\text{m}$ cell sizes is used for fine and narrower field of view. The vision system is shown in Fig. 14a.

A typical visually guided manipulation requires the position information of the object and the probe in x , y and z to realize 3D manipulation tasks. However, it is challenging to get the measurement in z axis for a micromanipulation task, since optical microscopes suffer from the low depth of field which limits the focal plane into a small range and causes defocused view of the object monitored outside this region. Although exploiting defocus can yield a coarse information along the z axes, it results in poor accuracy for micron precision applications and it is computationally expensive. Thus, to acquire the height information for the interested object, a lateral microscope with an additional CCD camera

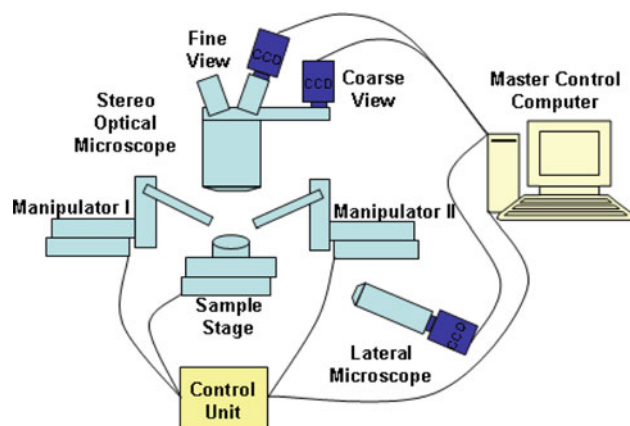
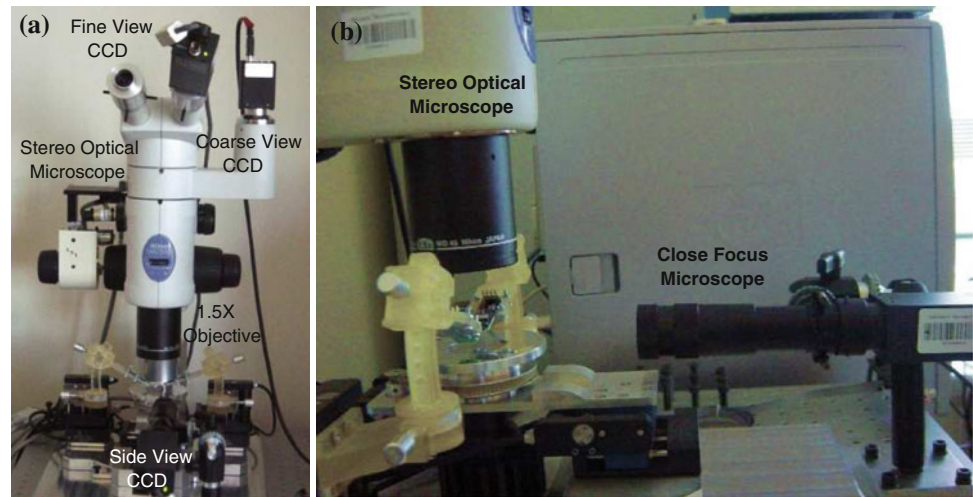
**Fig. 13** Entire model of workstation

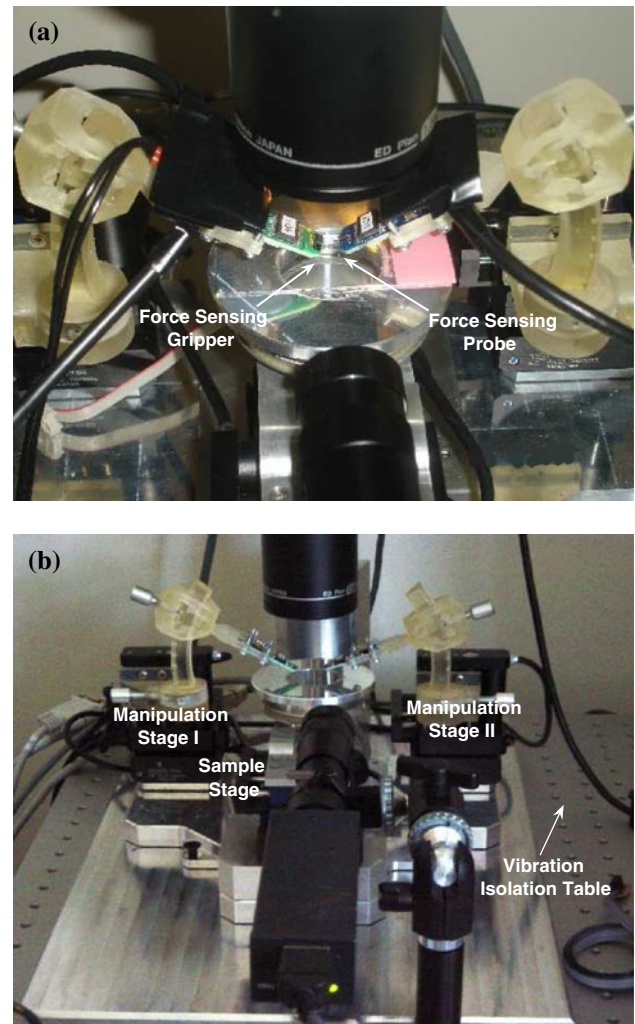
Fig. 14 Vision hardware of workstation

is employed. A Sony XCD-X710CR coupled to a $35\times$ close focus microscope with variable zoom and relatively long working distance is used to acquire the height information between the sample stage and the probe as shown in Fig. 14b.

It is also important to accurately handle the microparts for a dexterous manipulation, since the microparts to be manipulated are usually fragile. Therefore two types of end effectors, a probe and a microgripper, are integrated with capacitive force sensors and are able to sense the forces down to 0.4 and 0.01 μN , respectively. The force-sensing probe and gripper are mounted on tilted holders to reach the desired point effectively. An illustrative figure is shown in Fig. 15a. The inclined probes can approach microsamples without touching undesired points on the sample stage. However, the inclination leads to the requirement of a relatively large working distance and depth of field. Therefore, a $1.5\times$ objective with 44.5 mm working distance is utilized to provide well-focused images of the probes and samples.

Transporting and positioning the microparts are crucial for the execution of the microassembly tasks as well as the handling issues. The force-sensing probe and gripper are mounted on two separate 3-DOF fine positioning stages (PI M111.1DG with effective $x-y-z$ range of $15 \times 15 \times 15$ mm and 50 nm closed loop precision). On an $x-y-\theta$ positioning stage, (PIM111.1DG with effective $x-y$ range of 15×15 mm, 50 nm close loop precision, and $4.5 \times 10^{-5}^\circ$ rotation resolution) a glass slide is mounted and is positioned under the force sensing probe and microgripper. On the glass slide, the samples, polystyrene balls and biological cells, which are used in experiments, can be located. The high precision positioning stages are depicted in Fig. 15b.

The system consists of three IEE1394 cameras, 9 DC motors and an illumination system. All those components are controlled with a centralized structure including a main and a slave control computer. While the main control

**Fig. 15** Microassembly workstation. **a** Probes, **b** sample and manipulator stages

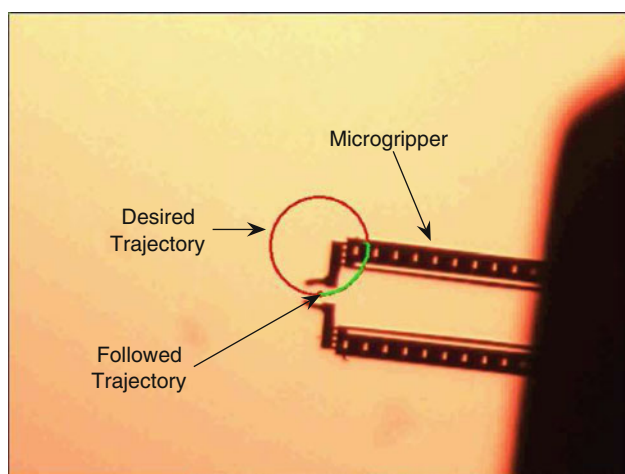


Fig. 16 Circular path following task at 1×

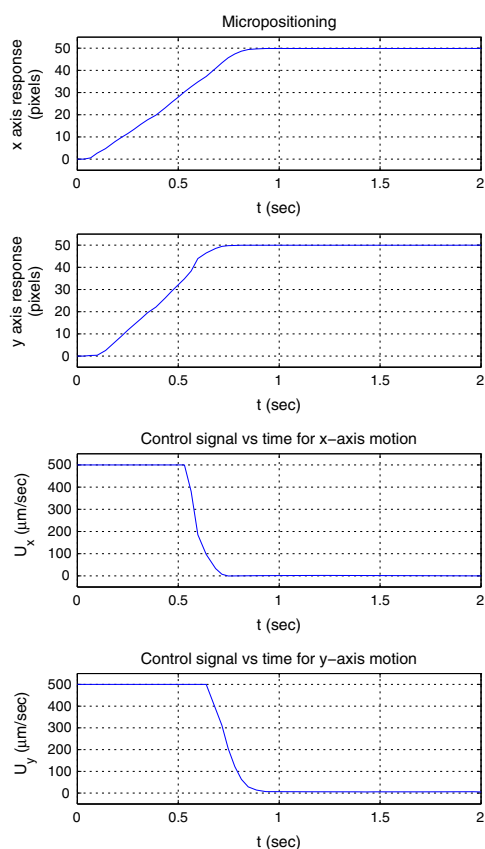


Fig. 17 Step responses and control signals at 1×

computer gives reference values without any direct connections to the actuators, force and positioning sensors, the slave one runs a fast control loop based on the directives from the main computer. The control software is written in the C programming environment and provides real-time control for moving and positioning the stages precisely. Moreover, video data from the IEE1394 cameras is directly transferred to the main computer and the computer vision soft-

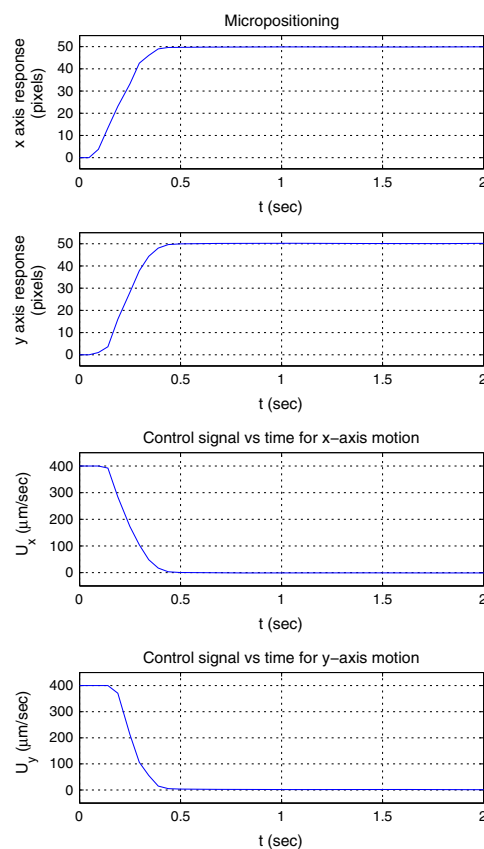


Fig. 18 Step responses and control signals at 4×

ware is written in C++ environment by using the OpenCV library.

Before presenting the microobject manipulation experiments, the experimental results using only vision feedback without touching any microobjects are shown to validate that visual servoing ensures the required accuracy and precision for the manipulation tasks.

5.2 Visual servoing experiments

To evaluate the performance of the presented visual servoing algorithm with the optimal control synthesis which penalizes the error and the control signal, regulation and path following experimental results are given. In the experiments, the center of the microgripper opening is tracked with subpixel accuracy at 30 Hz. Micropositioning and trajectory following tasks are performed at 1× and 4× zoom levels. An illustrative figure is depicted in Fig. 16. For the optimal control design, Q and R matrices in Eq. (25) were chosen as diagonal matrices with diagonal entries (0.9, 0.9) and (0.025, 0.05), respectively. Micropositioning visual servoing results are plotted in Figs. 17, 18, and the trajectory following results for circular, square and sinusoidal trajectories are depicted in Figs. 19, 20, 21.

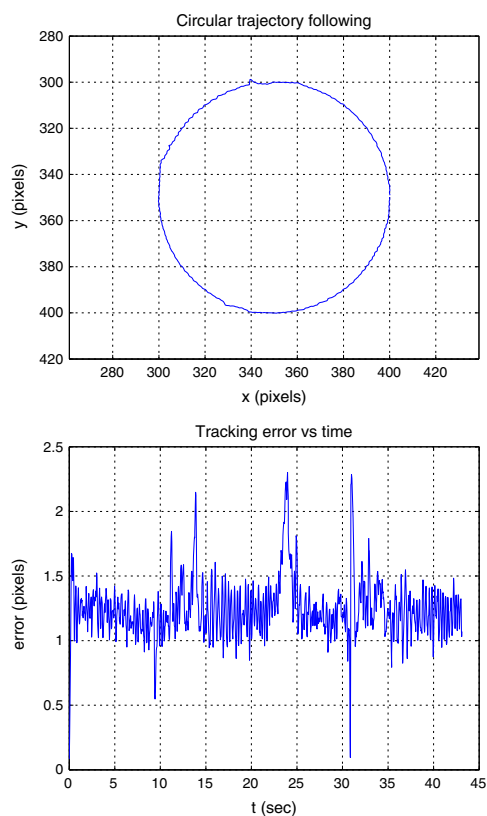


Fig. 19 Circular trajectory and tracking error at $1\times$

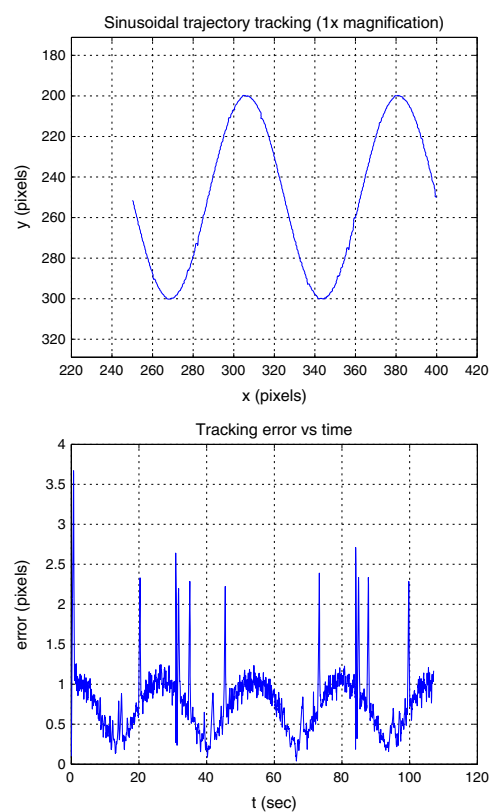


Fig. 21 Sinusoidal trajectory and tracking error at $1\times$

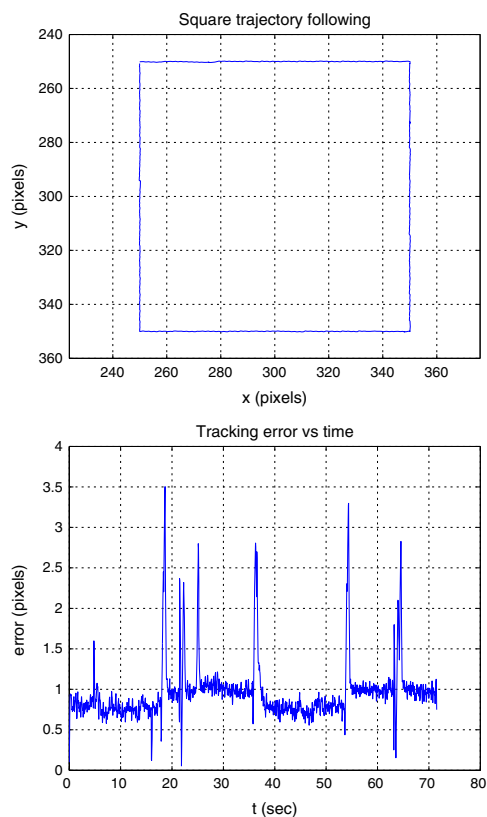


Fig. 20 Square trajectory and tracking error at $1\times$

Table 1 Micropositioning accuracy and precision

	Step (pixels)	t_s (sec)	Acc. (μm)	Prec. (μm)
$1\times$	50	0.80	9.86	2.71
$4\times$	50	0.45	1.35	0.57

For the micropositioning task, regulation performances at $1\times$ and $4\times$ for a step input of 50 pixels along the x and y axis of the objective frame in terms of settling time (t_s), accuracy (i.e., average positioning/tracking error, which implies proximity of the actual position value to the true value) and precision (i.e., standard deviation of the positioning/tracking errors, which implies repeatability of the measurements) are tabulated in Table 1. For the trajectory following task, tracking performances for different trajectories (square, circular and sinusoidal) are presented in Table 2.

The presented visual servoing guarantees convergence to the desired targets with sub-micron error and satisfactory settling time. Moreover, the tracking performance also meets the requirements for a typical micromanipulation task, which tolerates a few micron errors.

The spikes in Figs. 19–21 in the measured tracking errors may be due to the blurring edges of the gripper as a result of changing illumination angle with the moving gripper.

Table 2 Trajectory tracking accuracy and precision

	Square		Circular		Sinusoidal	
	Acc. (μm)	Prec. (μm)	Acc. (μm)	Prec. (μm)	Acc. (μm)	Prec. (μm)
$1\times$	5.93	2.28	7.72	1.40	4.79	2.37
$4\times$	1.47	1.19	1.57	0.95	1.12	1.31

One can tune the optimal control parameters (Q and R matrices in Eq. 25) and get a relatively slow motion with smaller tracking errors. This way, template matching can be constrained to a smaller region and local minima can be avoided. We are currently working on a new template-matching algorithm, which takes into account blur in the captured image. Estimating blur for each image and parameterizing the template model may ensure better performance in tracking.

5.3 Micromanipulation experiments

To validate the proposed collision-free micromanipulation method, polystyrene balls are pushed on the sample stage to the desired locations. Before the experiments, the polystyrene balls on the glass surface are scattered by evaporating the water in the prepared diluted polystyrene–water solution. Distilled water is preferred in the solution to prevent the contamination of unwanted substances on the substrate surface. In the experiments, a tipless AFM probe is chosen to push the individual microballs. After the user determines the target locations, automatic micromanipulation algorithm starts. In the following experiment, the operator chooses a pattern in which the centers of three microspheres are on a same straight line.

In the first step, the probe moves to the computed position in the z direction to push the microsphere from a convenient contact point, as shown in Fig. 22.

In the following steps, the microballs are pushed to the targets in the generated pattern using the motion-planning method described in Sect. 3. A summary of the experiment that illustrates the steps of the automatic micromanipulation task is given in Fig. 23.

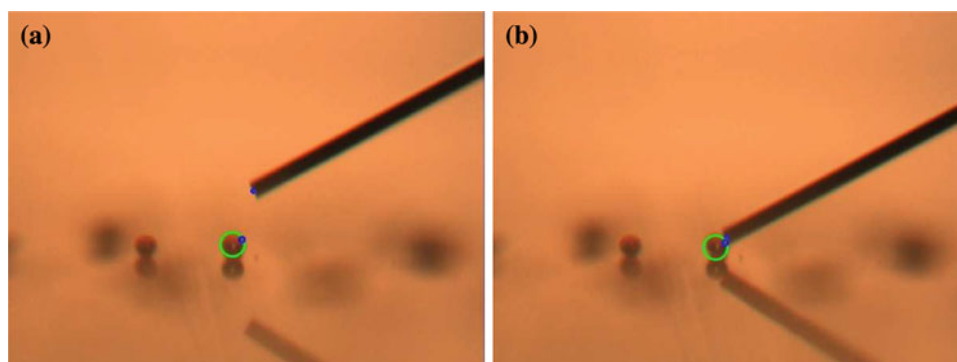
In the experiments, the $70\text{ }\mu\text{m}$ diameter microspheres can be transported to the desired locations with the accuracy of one pixel, which is the tolerance defined by the user. The accuracy of the pushing experiments can be improved to $0.7\text{ }\mu\text{m}$, which is the resolution of the optical microscope, using the presented subpixel detection algorithms. Therefore, a micromanipulation experiment in the microassembly workstation with the designed mechanical and software components is successfully demonstrated.

6 Conclusion

It has been shown that fully automated micromanipulation can be done at the precision of micron level using visual feedback. To compensate the errors due to the uncertainties about the position and shape of the microobjects to be manipulated, a robust vision control structure is proposed and implemented. To realize the closed-loop structure, auto-focusing, real-time object and end effector detection, tracking and optical system calibration modules are developed and integrated with visual controller and motion planner subsystems.

In a future study, motorized vision sensors can be integrated to the system to eliminate the disadvantages of small depth of field. Introducing new mobile vision sensors can maximize the resolvability by ensuring focused and nonoccluded views in micromanipulation tasks.

Fig. 22 Visual servoing in the x – z Plane



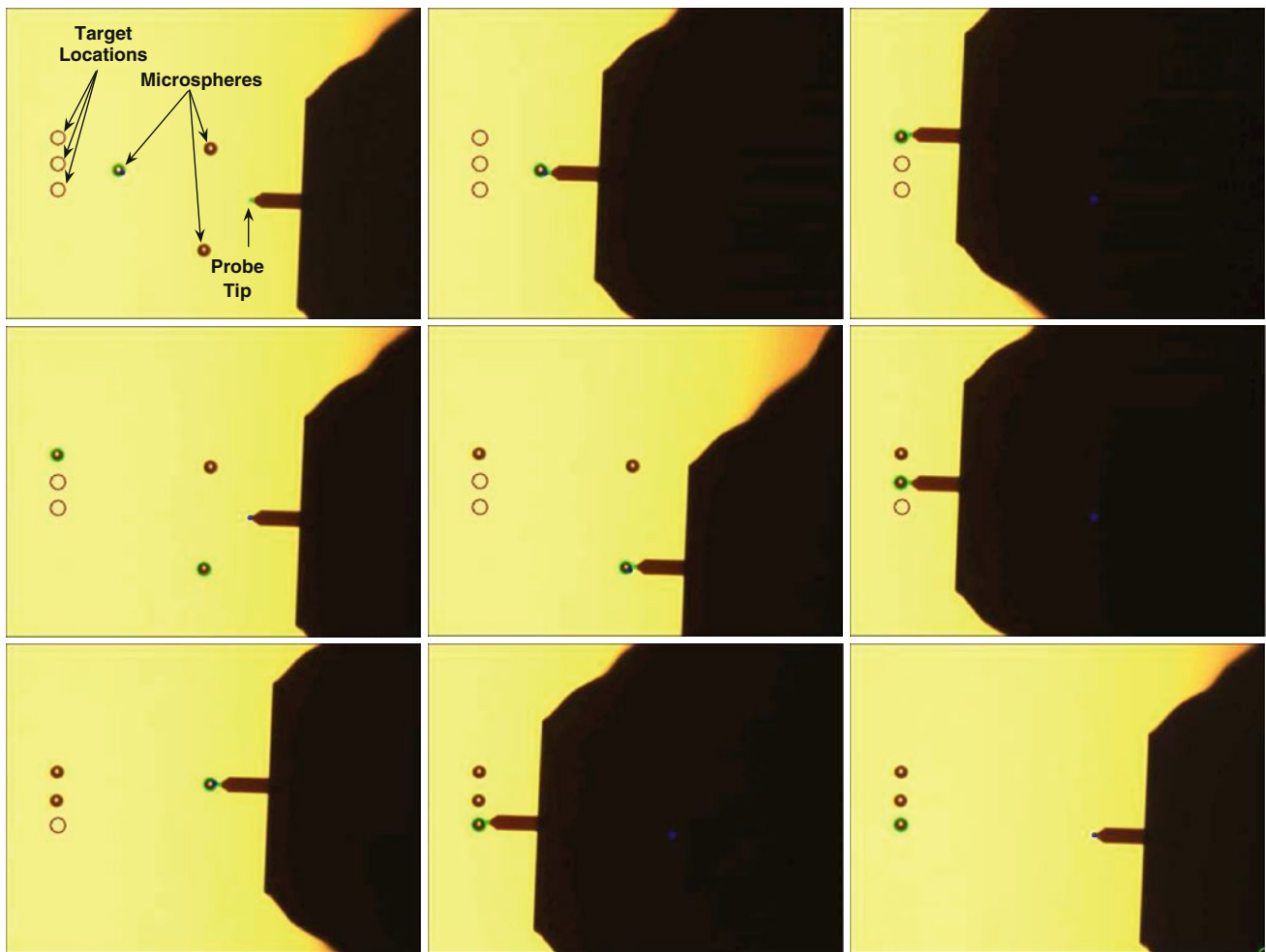


Fig. 23 Automatic micromanipulation of microspheres

Acknowledgments This work was supported by the SU Internal Grant No. IACF06-00417.

References

1. Boissenin, M., Wedekind, J., Selvan, A.N., Amavasai, B.P., Caparrelli, F., Travis, J.R.: Computer vision methods for optical microscopes. *Image Vis Comput* **25**(7), 1107–1116 (2006)
2. Amavasai, B.P., Caparrelli, F., Selvan, A., Boissenin, M., Travis, J.R., Meikle, S.: Machine vision methods for autonomous micro-robotic systems. *Kybernetes Int J Syst Cybern* **34**(9–10), 1421–1439 (2005)
3. Kim, B., Kang, H., Kim, D.H., Park, G.T., Park, J.O.: Flexible microassembly system based on hybrid manipulation scheme. In: *IEEE/RSJ Int. Conf. on intelligent robots and systems*, Nevada, pp. 2061–2066 (2003)
4. Enikov, E.T., Minkov, L.L., Clark, S.: Microassembly experiments with transparent electrostatic gripper under optical and vision-based control. *IEEE Trans. Ind. Electron.* **52**(4), 1005–1012 (2005)
5. Yang, G., Gaines, J.A., Nelson, B.J.: A flexible experimental work-cell for efficient and reliable wafer-level 3D micro-assembly. *IEEE Int. Conf. Robot. Automat* **1**, 133–138 (2001)
6. Yang, G., Gaines, J.A., Nelson, B.J.: Optomechatronic design of microassembly systems for manufacturing hybrid microsystems. *IEEE Trans. Ind. Electron.* **52**(4), 1013–1023 (2005)
7. Nelson, B.J., Zhou, Y., Vikramaditya, B.: Sensor-based micro-assembly of hybrid MEMS device. In: *IEEE Control Systems*, pp. 35–45 (1998)
8. Beyeler, F., Bell, D.J., Nelson, B.J., Sun, Y., Neild, A., Oberti, S., Dual, J.: Design of a micro-gripper and an ultrasonic manipulator for handling micron sized objects. In: *Int. Conf. on Intelligent Robots and Systems*, Beijing, China, pp. 772–777 (2006)
9. Dechev, N., Cleghorn, W.L., Mills, J.K.: Microassembly of 3-D MEMS structures utilizing a MEMS microgripper with a robotic manipulator. *IEEE Int. Conf. Robot. Automat.* **3**, 3193–3199 (2003)
10. Popa, D.O., Kang, B.J., Sin, J., Zou, J.: Reconfigurable micro-assembly system for photonics applications. In: *IEEE Int. Conf. on Robotics and Automation*, pp. 1495–1500, Washington (2006)
11. Sun, Y., Nelson, B.J.: Microrobotic cell injection. *Int. Conf. Robot. Automat.* **1**, 620–625 (2001)
12. Sun, Y.W., Kenneth, R.P., Bischof, J.C., Nelson, B.J.: Mechanical property characterization of mouse zona pellucida. *IEEE Trans. Nanobiosci.* **2**(4), 279–286 (2003)
13. Li, X., Zong, G., Bi, S.: Development of global vision system for biological automatic micromanipulation system. In: *Int. Conf. on Robotics and Automation*, Seoul, pp. 127–132 (2001)

14. Kim, D.H., Hwang, C.N., Sun, Y., Lee, S.H., Kim, B., Nelson, B.J.: Mechanical analysis of chorion softening in prehatching stages of zebrafish embryos. *IEEE Trans. Nanobiosci.* **5**(2), 89–94 (2006)
15. Wang, W.H., Liu, X.Y., Sun, Y.: Autonomous Zebrafish Embryo injection using a microrobotic system. In: *Int. Conf. on Automation Science and Engineering*, pp. 363–368 (2007)
16. Kasaya, T., Miyazaki, H., Saito, S., Sato, T.: Micro object handling under SEM by vision-based automatic control. In: *IEEE Int. Conf. Robotics and Automation*, Detroit, pp. 2189–2196 (1999)
17. Sitti, M.: Controlled pushing of nanoparticles: modeling and experiments. *IEEE/ASME Trans. Mechatron.* **5**(2), 199–221 (2000)
18. Krotkov, E., Martin, J.P.: Range from Focus. *IEEE Int. Conf. Robot. Automat.* **3**, 1093–1098 (1986)
19. Sun, Y., Duthaler, S., Nelson, B.J.: Autofocusing algorithm selection in computer microscopy. In: *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pp. 70–76 (2005)
20. Xiong, Y., Schafer, S.A.: Depth from Focusing and Defocusing. In: *IEEE Computer Society Conf. on computer vision and pattern recognition*, pp. 68–73 (1993)
21. Ma, Y., Soatto, S., Kosecka, J., Sastry, S.S.: *An Invitation to 3-D Vision, from Images to Geometric Models*. Springer, Berlin (2006)
22. Forsyth, D.A., Ponce, J.: *Computer Vision, a Modern Approach*. Prentice-Hall, Englewood Cliffs (2003)
23. Bilen, H., Hocaoglu, M., Unel, M., Gozuacik, D.: Novel parameter estimation schemes in microsystems. In: *IEEE Int. Conf. on Robotics and Automation*, pp. 2394–2399 (2009)
24. Tsai, R.Y.: A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE J. Robot. Automat.* **3**, 323–344 (1987)
25. Zhang, Z.Y.: Flexible camera calibration by viewing a plane from unknown orientations. In: *IEEE Int. Conf. on Computer Vision*, pp. 666–673 (1999)
26. Zhou, Y., Nelson, B.J.: Calibration of a parametric model of an optical microscope. *Optical Eng.* **38**, 1989–1995 (1999)
27. Ammi, M., Fremont, V., Ferreira, A.: Flexible microscope calibration using virtual pattern for 3-D telemicromanipulation. In: *IEEE Trans. on Robotics and Automation*, pp. 3888–3893 (2005)
28. Chaumette, F., Hutchinson, S.: Visual servo control I. Basic approaches. *IEEE Robot Automat Mag.* **13**(4), 82–90 (2006)
29. Wang, J., Cho, H.: Micropeg and hole alignment using image moments based visual servoing method. *IEEE Trans. Ind. Electron.* **55**(3), 1286–1294 (2008)
30. Motai, Y., Kosaka, A.: Hand-eye calibration applied to view-point selection for robotic vision. *IEEE Trans. Ind. Electron.* **55**(10), 3731–3741 (2008)
31. Choi, K.J., Lee, Y.H., Moon, J.W., Park, C.K., Harashima, F.: Development of an automatic stencil inspection system using modified hough transform and fuzzy logic. *IEEE Trans. Ind. Electron.* **54**(1), 604–611 (2007)
32. Malis, E.: Visual servoing invariant to changes in camera intrinsic parameters. *IEEE Trans Robot Automat* **20**, 72–81 (2004)
33. Gangloff, J., Mathelin, M., Abba, G.: 6 DOF high speed dynamic visual servoing using GPC controllers. In: *IEEE Int. Conf. on Robotics and Automation*, pp. 2008–2013 (1998)
34. Nelson, B.J., Papanikolopoulos, N.P., Khosla, P.K.: Robotic visual servoing and robotic assembly tasks. *IEEE Robot. Automat. Mag.* **3**(2), 23–31 (1996)
35. Hashimoto, K., Ebine, T., Sakamoto, K., Kimura, H.: Full 3D visual tracking with nonlinear model-based control. In: *American Control Conference*, San Francisco, pp. 3180–3184 (1993)