Stereovision-based Control for Automated MOEMS Assembly

Andrey V. Kudryavtsev, Guillaume J. Laurent, Cédric Clévy, Brahim Tamadazte and Philippe Lutz

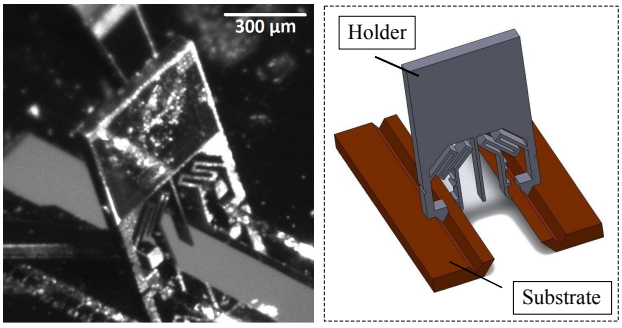
***Abstract*— Microassembly represents a very promising solution to microproducts and complex Micro-Electro-Mechanical Systems (MEMS) fabrication. Since, in the case of teleoperated assembly, an operator is the main source of errors, there is a great interest in microassembly automation. Its main issue consists in precise estimation of object position. Previous studies demonstrate the possibility of application of model-based visual tracking algorithms from ViSP (Visual Servoing Platform) library. However, the methods of macroassembly cannot be directly applied when working with microobjects. The characterization of single-view visual tracking notably revealed the complexity of depth estimation in microscale, which is due to small depth variation in the seen images compared with the distance from camera. So, an algorithm of Z coordinate reconstruction using a second camera was developed and analyzed for visual servoing task. It was then used to automate microassembly. Experiments demonstrate the possibility of complex microcomponent automatic microassembly with precision better than 10 micrometers.**

I. Introduction

Over the past two decades the tendency for objects miniaturization demonstrates a scientific interest toward complex and high precision MEMS [1]–[3]. These systems find their application in a great number of scientific fields such as biomedical engineering, aerospace manufacturing and instrumentation. The main stumbling block to MEMS development is the complexity of microfabrication process. Microassembly using microrobotic systems represents one of the alternative solutions to this problem.

Several works already established the viability of this approach and notably shown that the key feature relies in the capability of the system to achieve modular and highly accurate assembling, i.e., typically smaller than 5 μm (maximum acceptable error) [4]. In the presented work, a microoptical system displayed in Fig. 1 is used as a case of study. It represents an example of MEMS where the high assembly precision is crucial. Several concepts of microoptical benches to be assembled have also been proposed [5]–[7]. In [8], it is also established that a positioning accuracy smaller than 1 μm can be achieved in teleoperated mode. The operator being the main source of inaccuracies [8], there is a great interest in automating the microassembly process. It implies an increase of the throughput yield but also quantification of the main sources of inaccuracies which is of great interest for the design of MOEMS (Micro-Opto-Electro-Mechanical Systems) blocks and microrobots, the clean room fabrication and assembly strategies.

A. V. Kudryavtsev, G. J. Laurent, C. Clévy, B. Tamadazte and P. Lutz are with the AS2M Department FEMTO-ST Institute, UMR CNRS 6174 - UBFC / UFC / ENSMM / UTBM. 25000 France.
andrey.kudryavtsev at femto-st.fr



(a) (b)

Fig. 1: Example of assembled microoptical bench (a) and its CAD-model (b).

Previous works on the field of automatic microassembly demonstrate the possibility of model-based visual trackers application [9]. However, these strategies cannot be employed directly in our case, because of several constraints. First, all objects of the scene are made of silicon which causes reflections. Secondly, the object contains very small flexible parts (10×50 μm^2 of cross section) which come into contact with microgripper for object manipulation. Finally, the ratio between length and thickness is very high. The preliminary work on the characterization of single-view visual tracking techniques was conducted [10]. The obtained results prove that it is possible to have a precision better than 1 μm for X and Y coordinates in camera frame. However, the depth coordinate cannot be correctly estimated in microscale due to the fact that the focal distance is much bigger than sensor size, which results in the situation where projective rays become parallel. One of the possible solutions consists in installation of the second camera in the plane perpendicular to the Z axis of the first camera, which will allow to reconstruct the missing information about the depth coordinate. Therefore, the main goal of this paper consists in estimation of object position in stereo setup and applying stereovision-based control to automate the assembly process.

A first step consists in precise estimation of 3D object position (XYZ) using stereo visual feedback to achieve automated assemblies of MOEMS and then study the viability and precision of this approach at the microscale. To achieve this goal, a strategy based on high level closed-loop vision control will be implemented. The studied methods are model-based visual tracking algorithms from the ViSP library [11], which is able to directly provide the 3D object pose using a single view of the scene.

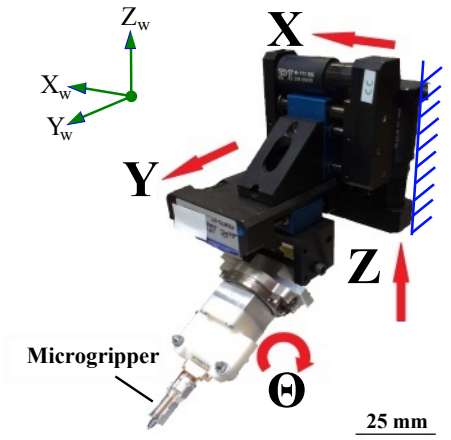


Fig. 2: 4-DOF robot with microgripper.

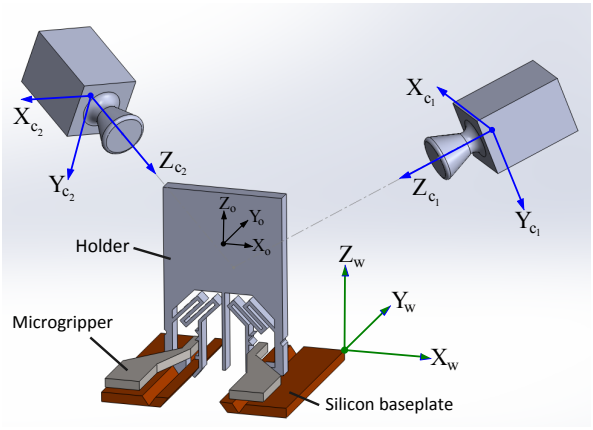


Fig. 3: Cameras positions in stereo setup.

The remainder of this paper is organized as follows: Section II presents the equipment used, micromanipulation station in particular, and reminds the results of single-view tracking characterization. In Section III, we present the approach of the depth coordinate reconstruction, that will be used further in Section IV, which describes visual servoing control law and analyses its results. Section V represents an experimental validation of our work: assembly automation and its brief analysis. Finally, conclusions and prospects are discussed at the end.

II. Single View Model-Based Tracking

A. Experimental setup

For an accurate control of position and alignment of optical path, we use the 3D microassembly station that comprises a serial robot of 4 degrees of freedom (XYZΘ) with a 4-DOF microgripper (Fig. 2) and a vision system (Fig. 3). The whole system is placed on an antivibration table. The characteristics of the robot and vision system are represented in Tables I and II, respectively.

TABLE I: Characteristics of the stages comprised in the robot used in a micromanipulation station.

	Reference	Specifications
Translation stages : XY	M-111-DG *PI Mercury*	Stroke : 15 mm Backlash : 2 µm Min. Inc. Motion : 0.05 µm Unidir. repeatability : 0.1 µm
Z	M-121-DG *PI Mercury*	Stroke : 25 mm Backlash : 2 µm Min. Inc. Motion : 0.05 µm Unidir. repeatability : 0.1 µm
Rotation stage : Θ	SR3610S *SmarAct*	Stroke : 360° Resolution : $< 10\mu$ °

TABLE II: Characteristics of the vision system.

	Reference	Specifications
2x Cameras : IDS uEye	UI-3480CP *Aptina*	CMOS Rolling Shutter Pixelpitch : 2.2 µm Pixel Class : 5 Megapixel Resolution (h x v) : 2560x1920
Objectif	CVO GM10HR35028	Class : high resolution Focal distance : 50 mm

B. Single View Model-Based Tracking Characterization

The performance of the single view model-based visual tracking in microscale have been previously estimated in [10]. The latter concentrates on the estimation of noises influencing the tracking by conducting two types of experiments:

- Tracker analysis with a static object (the object pose is recorded while the robot is not moving);
- Planned path tracking (comparing the measurements between proprioceptive robot sensors and the visual tracker).

The experiments for the analysis with a static object and planned path following give the results presented in Tables III and IV. In the case of static object tracking, one can notice that the standard deviation of position measurement for all of the trackers along X and Y axes of the camera does not exceed 1 µm which is a very promising result in terms of its further application for the assembly tasks. For the planned path tracking, the standard deviations of errors between the tracker and the robot joint coordinates attain 2.8 µm along X and 4 µm along Y axis. These deviations include intrinsic robot positioning errors which can typically reach several micrometers of amplitude and can be compensated by visual servoing control law since closed-loop control is used. The depth coordinate cannot be correctly estimated, which is due to the fact that if the set of points seen in the image has small depth variation compared with the distance from the camera, the projection model becomes close to a parallel one [12]. Thus, it is more difficult to estimate the z

TABLE III: Standard deviation of position measurement for different trackers with a static object in the camera frame.

Coordinates	Edge-based [13]	Texture-based [14]	Hybrid [15]
X	0.2245 µm	0.7861 µm	0.2551 µm
Y	0.7026 µm	0.8650 µm	0.6207 µm
Z	**24.3304 µm**	**44.1736 µm**	**14.8640 µm**
roll	0.0859°	0.1607°	0.0610°
pitch	0.0577°	0.1254°	0.0539°
yaw	0.0608°	0.0984°	0.0461°

TABLE IV: Standard deviation error obtained between visual hybrid tracker and robot sensors in camera frame.

Coordinates	Standard deviation error
Δx	2.8165 µm
Δy	4.0918 µm
Δ**z**	**215.8632 µm**

coordinate in camera frame. This problem particularly shows up in microscale when one use a high magnification vision system (optical microscope) and demonstrates the necessity of supplementary sensor usage, i.e., second camera.

III. Depth Estimation in Stereo-Vision Setup

The use of the second camera gives an additional information about object position and allows to reconstruct depth coordinate. The stereo vision system is placed in the way that Z axis of the first camera should be perpendicular to Z axis of the second camera. One of the important steps of depth estimation, in case of two cameras, is the calibration of the entire system, i.e., the estimation of the transformation matrices between different elements (robot sensors, first camera, second camera) in order to be able to express all data in a common frame. These matrices are schematically represented on Figure 4

Thereafter, we use the following notations:

- i image number;
- $\mathcal{R}_{c_1}, \mathcal{R}_{c_2}$ cameras frames;
- $\mathcal{R}_w$ world frame;
- ${}^{c_1}\mathbf{P}_o, {}^{c_2}\mathbf{P}_o$ the pose of the object in the camera frame obtained with the tracker,
$$ {}^{c}\mathbf{P}_o = \begin{pmatrix} {}^{c}\mathbf{R}_o & {}^{c}\mathbf{t}_o \\ 0 & 1 \end{pmatrix} $$
where ${}^{c}\mathbf{t}_o = ({}^{c}x_o \ {}^{c}y_o \ {}^{c}z_o)^\top$;
- ${}^{w}\overline{\mathbf{P}}_o$ the pose of the object in $\mathcal{R}_w$ (variables with overline are referred to values estimated using proprioceptive robot sensors), rotations are not taken into account,
$$ {}^{w}\overline{\mathbf{P}}_o = \begin{pmatrix} \mathbf{I}_{3\times3} & {}^{w}\overline{\mathbf{t}}_o \\ 0 & 1 \end{pmatrix} $$

A. Transformation between $\mathcal{R}_w$ and camera frames

In order to be able to reconstruct ${}^{c_1}z_o$, it is necessary to transform the object coordinates in the world frame ${}^{w}\overline{\mathbf{t}}_o$ (obtained with proprioceptive robot sensors) to the frame of each cameras, this for each image. Thereafter, in the cases where calculation is similar for both cameras, the indexes c_1 and c_2 are omitted and index c is used instead. The transformation can be represented by a matrix ${}^{c}\mathbf{M}_w$ which contains the information about frame rotations and translations

$$ {}^{c}\overline{\mathbf{P}}_o = {}^{c}\mathbf{M}_w \, {}^{w}\overline{\mathbf{P}}_o \tag{1} $$

where ${}^{c}\mathbf{M}_w$ is the homogeneous matrix, which represents the extrinsic parameters of the camera, is not known (for both cameras). In order to estimate ${}^{c}\mathbf{M}_w$, an optimization algorithm was used. The goal of such optimization consists in minimization of the distance $\mathbf{\Delta}$ between the pose obtained from the tracker ${}^{c}\mathbf{P}_o$ and the sensor values of robot axis ${}^{c}\overline{\mathbf{P}}_o$ in camera frame along a 3D planned path. So, for each image

$$ \mathbf{\Delta} = \begin{pmatrix} {}^{c}\mathbf{t}_o \\ 1 \end{pmatrix} - {}^{c}\mathbf{M}_w \begin{pmatrix} {}^{w}\overline{\mathbf{t}}_o \\ 1 \end{pmatrix} \tag{2} $$

The optimization criteria is then defined as a sum of squared distance between two curves as

$$ J = \sum_{i=1}^{n} \mathbf{\Delta}_i^T \, \mathbf{C} \, \mathbf{\Delta}_i \tag{3} $$

In order to minimize the influence of ${}^{c}z_o$ coordinate, the coefficient of 0.001 was applied.

$$ \mathbf{C} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.001 \end{pmatrix} $$

The used optimization algorithm is a Levenberg-Marquardt algorithm that is implemented in the Mathworks Optimization Toolbox [16]. It was programmed to take the best fit on 20 optimizations from random initial transforms. So, using this algorithm for each camera, the matrices ${}^{c_1}\mathbf{M}_w$ and ${}^{c_2}\mathbf{M}_w$ can be estimated.

B. Transformation between $\mathcal{R}_{c_1}$ and $\mathcal{R}_{c_2}$

Further calibration of the visual system represents the estimation of the transformation matrix ${}^{c_2}\mathbf{M}_{c_1}$ between cameras frames. This matrix allows to represent the object position obtained with the first camera ($\mathcal{R}_{c_1}$) in $\mathcal{R}_{c_2}$. This matrix can be found using world frame ($\mathcal{R}_w$) as an intermediary stage, that gives the following equation:

$$ {}^{c_2}\mathbf{M}_{c_1} = {}^{c_2}\mathbf{M}_w \, {}^{w}\mathbf{M}_{c_1} = {}^{c_2}\mathbf{M}_w ({}^{c_1}\mathbf{M}_w)^{-1} \tag{4} $$

It has to be highlighted, that it is impossible to find the matrix ${}^{c_2}\mathbf{M}_{c_1}$ directly using the same algorithm as for ${}^{c_1}\mathbf{M}_w$ and ${}^{c_2}\mathbf{M}_w$, because the information about z coordinate is wrong for the poses obtained with the camera c_1 as well as with the camera c_2.

So, knowing ${}^{c_1}\mathbf{M}_w$ and ${}^{c_2}\mathbf{M}_w$, one can easily find ${}^{c_2}\mathbf{M}_{c_1}$ matrix.

C. Depth Estimation

The relation between the object coordinates in $\mathcal{R}_{c_1}$ and $\mathcal{R}_{c_2}$ can be written as follows

$$ \begin{pmatrix} {}^{c_2}\mathbf{t}_o \\ 1 \end{pmatrix} = {}^{c_2}\mathbf{M}_{c_1} \begin{pmatrix} {}^{c_1}\mathbf{t}_o \\ 1 \end{pmatrix} \tag{5} $$

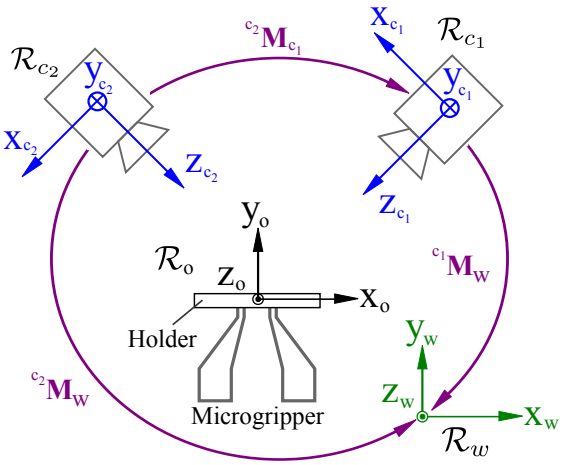


Fig. 4: Representation of transformation matrices.

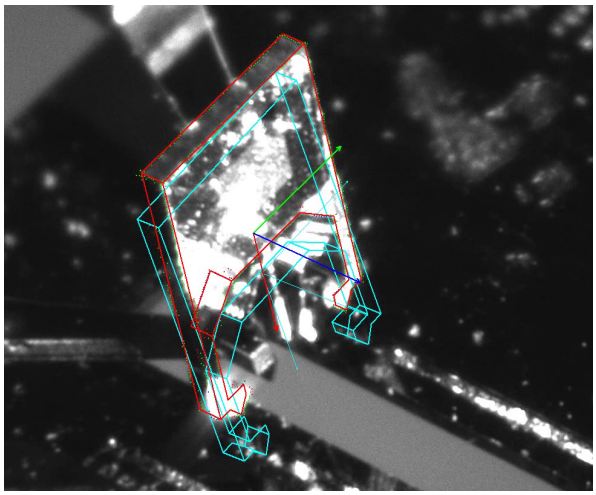


Fig. 5: Image acquired with the camera c_1 with current pose (red) and desired pose (blue).

More in details

$$\begin{pmatrix} {}^{c_2}x \\ {}^{c_2}y \\ {}^{c_2}z \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} {}^{c_1}x \\ {}^{c_1}y \\ {}^{c_1}z \\ 1 \end{pmatrix} \tag{6}$$

In order to eliminate the unknown term $({}^{c_2}z)$, we use a matrix of orthographic projection $\mathbf{P}$, so that

$$\mathbf{P} \begin{pmatrix} {}^{c_2}\mathbf{t}_o \\ 1 \end{pmatrix} = \mathbf{P}\, {}^{c_2}\mathbf{M}_{c_1} \begin{pmatrix} {}^{c_1}\mathbf{t}_o \\ 1 \end{pmatrix} \tag{7}$$

with

$$\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

By decomposing ${}^{c_1}\mathbf{t}_o$ we obtain:

$$\mathbf{P} \begin{pmatrix} {}^{c_2}x \\ {}^{c_2}y \\ {}^{c_2}z \\ 1 \end{pmatrix} = \mathbf{P}({}^{c_2}\mathbf{M}_{c_1}) \left(\begin{pmatrix} {}^{c_1}x \\ {}^{c_1}y \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ {}^{c_1}z \\ 0 \end{pmatrix} \right) \tag{8}$$

$$\underbrace{\mathbf{P} \begin{pmatrix} {}^{c_2}x \\ {}^{c_2}y \\ {}^{c_2}z \\ 1 \end{pmatrix} - \mathbf{P}({}^{c_2}\mathbf{M}_{c_1}) \begin{pmatrix} {}^{c_1}x \\ {}^{c_1}y \\ 0 \\ 1 \end{pmatrix}}_{\mathbf{B}} = \underbrace{\mathbf{P}({}^{c_2}\mathbf{M}_{c_1}) \begin{pmatrix} 0 \\ 0 \\ {}^{c_1}z \\ 0 \end{pmatrix}}_{\mathbf{A}\,({}^{c_1}z)} \tag{9}$$

$$\mathbf{A}\, {}^{c_1}z = \mathbf{B} \tag{10}$$

Thus, we obtain an over-determined system with two equations and one unknown that can be found using the pseudo-inverse of the matrix $\mathbf{A}$ defined as $\mathbf{A}^+$ (which provides the optimal solution in a least-squares sense):

$${}^{c_1}\hat{z} = \mathbf{A}^+ \mathbf{B} \tag{11}$$

So, we obtain two equations with one unknown variable ${}^{c_1}\hat{z}$. Equation (11) is used in real time process that allows to have enough information on the object position to achieve microassembly tasks. When ${}^{c_1}z$ is reconstructed, the procedure of the Section III-A is repeated once again to re-estimate the matrix ${}^{c_1}\mathbf{M}_w$.

IV. Visual Servoing

Once the system calibration is done, it is possible to correctly measure the 3D position of the object. So, as the next step, the control law is to be developed.

A. Control Law

Among existing types of visual servoing, PBVS (Position Based Visual Servoing) was chosen for the following reasons: first, because using visual tracking techniques we obtain directly the 3D object position. Its main advantage consists in the fact that the set point for control loop can be expressed in Cartesian coordinate system. Secondly, using PBVS one obtain better robot behavior in Cartesian space contrary to Image Based Visual Servoing (in image plane). So, as an input of control loop we use a 3D pose, which represents the desired position of the object, $\mathbf{s}^*$. Current object position in every iteration will be noted as $\mathbf{s}(t)$. The both quantities are expressed in $\mathcal{R}_{c_1}$. A visual servoing control law consists in minimizing the error between the current pose $\mathbf{s}(t)$ and the desired pose $\mathbf{s}^*$ (Fig. 5):

$$\mathbf{e}(t) = \mathbf{s}(t) - \mathbf{s}^* \tag{12}$$

where

$$\mathbf{s}(t) = \begin{pmatrix} {}^{c_1}x_o(t) \\ {}^{c_1}y_o(t) \\ {}^{c_1}\hat{z}_o(t) \\ 1 \end{pmatrix}$$

The goal now is to find the relation between this error (12) and robot movement, i.e., to find the desired robot speed for every axis allowing to minimize $\mathbf{e}(t)$ and use it as control signal. In order to assure system stability in closed-loop, the following Lyapunov function was defined [17]

$$V(t) = \frac{1}{2}\|\mathbf{e}(t)\|^2 = \frac{1}{2}\mathbf{e}^\top \mathbf{e} \tag{13}$$

TABLE V: Final visual servoing errors for different values of λ.

Coordinate	$\lambda = 0.1$	$\lambda = 0.05$
Δx	-2.21 μm	**-1.68 μm**
Δy	-4.11 μm	**0.64 μm**
Δz	4.66 μm	**-1.09 μm**

The derivative of this function is

$$\dot{V}(t) = \mathbf{e}^\top \dot{\mathbf{e}} \tag{14}$$

A controller capable to impose $\dot{\mathbf{e}} = -\lambda \mathbf{e}$ with $\lambda > 0$ guarantees the asymptotic stability of closed-loop system because $\dot{V}(t)$ will be always negative. Then, it can be noticed that as for kinematic model of the robot, it is possible to find a linear relation between current pose changing speed $\dot{\mathbf{s}}$ and the robot speed $\boldsymbol{\xi}$, i.e., we can find a matrix $\mathbf{L_s}$ that

$$\dot{\mathbf{s}} = \mathbf{L_s}\boldsymbol{\xi} \tag{15}$$

It can be underlined that the matrix $\mathbf{L_s}$, which relates the time variation of $\mathbf{s}$ in camera frame to the end-effector velocity in the world frame, represents the ${}^{c_1}\mathbf{M}_w$ matrix knowing that Jacobian robot matrix is equal to identity for presented robot structure. In our case, robot speed is defined by the linear speed, as rotations are not taken into account:

$$\boldsymbol{\xi} = (v_x, v_y, v_z, 1)^\top \tag{16}$$

Using $\mathbf{L_s}$ matrix, we can calculate the derivative of the error:

$$\dot{\mathbf{e}}(t) = \frac{d}{dt}(\mathbf{s}(t) - \mathbf{s}^*) = \dot{\mathbf{s}}(t) = \mathbf{L_s}\boldsymbol{\xi} \tag{17}$$

By imposing $\dot{\mathbf{e}} = -\lambda \mathbf{e}$, we obtain:

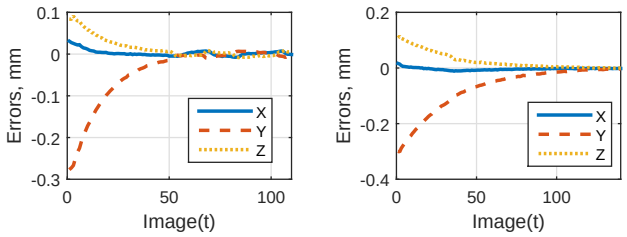
$$\boldsymbol{\xi} = \lambda {\mathbf{L_s}}^{-1} \mathbf{e} \tag{18}$$

The estimation of the ${}^{c_1}\mathbf{M}_w$ matrix was presented in Section III. So, in the final control law, joints speeds are calculated as

$$\dot{\mathbf{q}} = \boldsymbol{\xi} = -\lambda ({}^{c_1}\mathbf{M}_w)^{-1} \begin{pmatrix} {}^{c_1}\mathbf{t}_o - {}^{c_1}\mathbf{t}_o^* \\ 1 \end{pmatrix} \tag{19}$$

B. Results with Different Values of λ

The choice of gain λ depends mainly on the number of frames per second that camera can provide. In the case of higher gain, the difference between object position in two consecutive images becomes too important, so, the tracking algorithm will have some convergence problems. If it is too small ($\lambda \ll 0.01$), the displacement speed will decrease and the assembly will take much more time. Here, we provide the results for two values of λ: $\lambda = 0.1$ and $\lambda = 0.05$. They are represented in Fig. 6 and in Tab. V. The choice of λ value is a compromise between convergence speed and final precision. For further experiments, a value of 0.05 is used. The error in this case does not exceed 1.7 μm.



(a) λ = 0.1 (b) λ = 0.05

Fig. 6: Visual servoing errors for different values of λ.

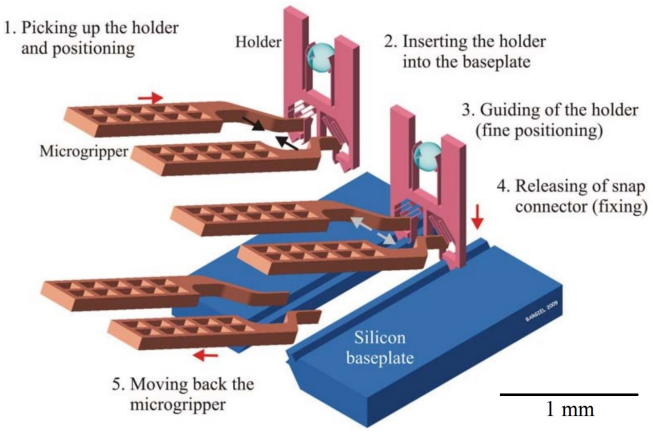


Fig. 7: General concept of holder assembly based on the use of robotic microgripper [5].

V. Experimental Validation

The final goal of our work was to automate the microassembly process of micro-optical bench and, thus, validate the effectiveness of presented approach of object position estimation in a real application where high precision is required. The assembly consists in insertion of holder in the V-groove guiding rails of silicon baseplate using a micromanipulation station (Fig. 7). The complexity of this task is due to a special form of holder: first, it contains very small flexible parts (span connectors) relative to the object size. Secondly, the holder thickness is about 20 times smaller than its height and weight. For holder positioning in the field-of-view of both cameras, a programmed point-to-point trajectory is used. This task doesn't require high precision, so, the proprioceptive robot sensors are used. On the other hand, the insertion of the holder in the base plate is the step which defines the quality of assembly and, at the same time, is the most complicated in the teleoperated mode, so the objective is to automate it using visual servoing. First of all, one should succeed in realizing an assembly in teleoperated mode (steps 1-5, Fig. 7) in order to define the desired pose $\mathbf{s}^*$ for further automated assemblies. Once it's done, a disassembly process is carried out, so an automated cycle begins from Step 2 until Step 4. After doing several consecutive automated assemblies, one can notice that the mean absolute positioning error (Fig. 8, Table VI) before

TABLE VI: Mean absolute errors of assembly before and after releasing the object.

Mean assembly error	Before releasing	After releasing
e_x	3.07 µm	9.88 µm
e_y	3.56 µm	8.91 µm
e_z	4.63 µm	6.00 µm

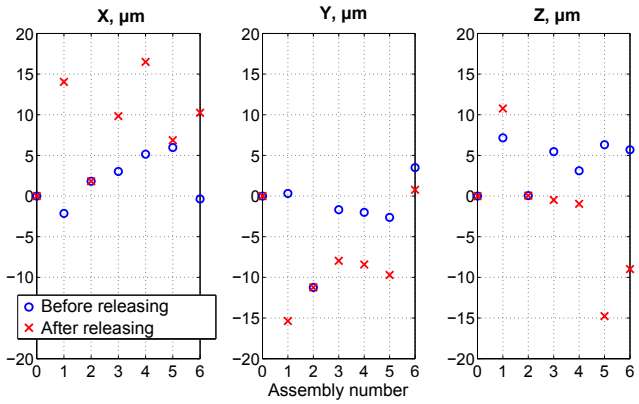


Fig. 8: Assembly errors due to the step of component release measured by visual system in $\mathcal{R}_w$.

holder releasing is inferior to 5 µm. The error becomes bigger compared to visual servoing error, because during assembly the object comes into contact with a silicon base plate that results in uncontrollable rotations of the object that cannot be compensated using our robot structure. The mechanical structure of the holder was developed in a sort that while releasing it can compensate angular position errors thanks to the particular form of the object and the base plate: when snap connector is released one can observe "fastening" between objects. However, after several experiments, we can notice that angular errors are compensated only partially. This effect of "fastening" also results in change of object position that explains the increasing errors (Fig. 8 and Table VI).

VI. Conclusions

The precise estimation of 3D position of an object is a key point of microassembly process. The problem of 3D object position estimation in microscale have been solved using stereo model-based visual tracking and applying a linear algorithm of depth coordinate reconstruction. This algorithm does not depend on object form and can be applied in all cases where CAD model can be defined. It uses the information from both cameras in order to estimate the depth and gives an optimal solution in a least-squared sense. The applied visual servoing control law gives a precision better than 2 µm. Finally, the experimental results on automatic microassembly prove the viability of presented approach: the mean absolute error of micropositioning is inferior to 5 µm. After component releasing the assembly error stays inferior to 10 µm. Assembly automation also allows to increase repeatability and decrease cycle time (from 10 minutes for teleoperated mode to about 15 seconds). The presented approach can be used for closed-loop control of MEMS, as well as a measurement tool in different microsystems.

Acknoledgment

These works have been funded by the Franche-Comté region, partially supported by the Labex ACTION project (contract "ANR-11-LABX-01-01"), by Labex ROBOPTIC project and by the French RENATECH network through its FEMTO-ST technological facility.

References

[1] D. Tolfree and M. J. Jackson, *Commercializing micro-nanotechnology products*. CRC Press, 2010.

[2] F. S. Chau, Y. Du, and G. Zhou, "A micromachined stationary lamellar grating interferometer for fourier transform spectroscopy," *Journal of Micromechanics and Microengineering*, vol. 18, no. 2, p. 025023, 2008.

[3] R. Syms, H. Zou, and J. Stagg, "Micro-opto-electro-mechanical systems alignment stages with vernier latch mechanisms," *Journal of Optics A: Pure and Applied Optics*, vol. 8, no. 7, p. S305, 2006.

[4] J. Agnus, N. Chaillet, C. Clévy, S. Dembélé, M. Gauthier, Y. Haddab, G. Laurent, P. Lutz, N. Piat, K. Rabenorosoa *et al.*, "Robotic microassembly and micromanipulation at femto-st," *Journal of Micro-Bio Robotics*, vol. 8, no. 2, pp. 91–106, 2013.

[5] S. Bargiel, K. Rabenorosoa, C. Clevy, C. Gorecki, and P. Lutz, "Towards micro-assembly of hybrid moems components on a reconfigurable silicon free-space micro-optical bench," *Journal of Micromechanics and Microengineering*, vol. 20, no. 4, p. 045012, 2010.

[6] A. N. Das, J. Sin, D. O. Popa, and H. E. Stephanou, "On the precision alignment and hybrid assembly aspects in manufacturing of a microspectrometer," *IEEE International Conference on Automation Science and Engineering*, pp. 959–966, 2008.

[7] K. Aljasem, L. Froehly, A. Seifert, and H. Zappe, "Scanning and tunable micro-optics for endoscopic optical coherence tomography," *IEEE Journal of Microelectromechanical Systems*, vol. 20, no. 6, pp. 1462–1472, 2011.

[8] C. Clévy, I. Lungu, K. Rabenorosoa, and P. Lutz, "Positioning accuracy characterization of assembled microscale components for micro-optical benches," *Assembly Automation*, vol. 34, no. 1, 2014.

[9] B. Tamadazte, E. Marchand, S. Dembélé, and N. Le Fort-Piat, "Cad model-based tracking and 3d visual-based control for mems microassembly," *The International Journal of Robotics Research*, 2010.

[10] A. V. Kudryavtsev, G. J. Laurent, C. Clévy, B. Tamadazte, and P. Lutz, "Characterization of model-based visual tracking techniques for moems using a new block set for matlab/simulink." in *IEEE International Symposium on Optomechatronic Technologies*, Seattle,USA, nov 2014, pp. 1–6.

[11] E. Marchand, F. Spindler, and F. Chaumette, "Visp for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robotics & Automation Magazine*, vol. 12, no. 4, pp. 40–52, 2005.

[12] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2003.

[13] A. Comport, E. Marchand, M. Pressigout, and F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *IEEE Trans. on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 615–628, July 2006.

[14] E. Marchand and F. Chaumette, "Feature tracking for visual servoing purposes," *Robotics and Autonomous Systems*, vol. 52, no. 1, pp. 53–70, July 2005.

[15] M. Pressigout and E. Marchand, "Real-time hybrid tracking using edge and texture information," *The International Journal of Robotics Research*, vol. 26, no. 7, pp. 689–713, 2007.

[16] S. Prajna, A. Papachristodoulou, and P. A. Parrilo, "Sostools: sum of squares optimization toolbox for matlab–users guide," *Control and Dynamical Systems, California Institute of Technology*, vol. 91125, 2004.

[17] F. Chaumette and S. Hutchinson, "Visual servo control. i. basic approaches," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 4, pp. 82–90, 2006.