

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Dynamic Compensation Framework to Improve the Autonomy of Industrial Robots

Shouren Huang, Yuji Yamakawa and Masatoshi Ishikawa

Abstract

It is challenging to realize the autonomy of industrial robots under external and internal uncertainties. A majority of industrial robots are supposed to be programmed by teaching-playback method, which is not able to handle with uncertain working conditions. Although many studies have been conducted to improve the autonomy of industrial robots by utilizing external sensors with model-based approaches as well as adaptive approaches, it is still difficult to obtain good performance. In this chapter, we present a dynamic compensation framework based on a coarse-to-fine strategy to improve the autonomy of industrial robots while at the same time keeping good accuracy under many uncertainties. The proposed framework for industrial robot is designed along with a general intelligence architecture that is aiming to address the big issues such as smart manufacturing, industrial 4.0.

Keywords: industrial robot, autonomy, intelligence architecture, high-speed sensing, coarse-to-fine strategy

1. Introduction

Performance of industrial robots in realizing fast and accurate manipulation is very important for manufacturing process, as it directly relates to productivity and quality. On the other hand, with manufacturing shifting from an old era of mass production to a new era of high-mix low-volume production, autonomous capability of industrial robots becomes more and more important to the manufacturing industry. Autonomy represents the ability of a system in reacting to changes and uncertainties on the fly.

Currently, off-line teaching-playback using a teaching pendant, or physically positioning a robot with a teaching arm, is supposed to be the main method for the applications of industrial robots. The method features a user-friendly interface developed by commercial robot manufacturers and is usually motion optimized and reliable so long as task conditions do not change. As detailed in [1], negative effects of nonlinear dynamics during high-speed motion may be pre-compensated in order to achieve accurate path tracking during the playback phase. However, it is impossible for a teaching-playback robot to adapt to significant variations in the initial pose of a working target or unexpected fluctuations during manipulation. CAD model-based teaching methods neither enable a robot to adapt to changes on the fly.

In [2], a view-based teaching-playback method was proposed to achieve robust manipulation against changes in task conditions with the use of artificial neural network (ANN). However, the approach is difficult for teaching jerky robot motion and cannot be applied to cases where high motion accuracy is required.

By utilizing external sensory feedback (e.g., vision), on-line control method may help a robot adjust to environmental uncertainty. Generally, accurate models with structured working environment are preconditions for implementation. However, in reality, accurate models are difficult to obtain. Regarding these issues, many adaptive approaches have been proposed (e.g., [3–7]) to address the control problem in the presence of uncertainty associated with a robot's kinematics model, mechanical dynamics, or with sensor-robot mapping. However, it is usually difficult to obtain satisfactory accuracy at a fast motion speed due to the complex dynamics and large mechanical inertia of a typical multi-joint industrial robot [8, 9].

In order to improve autonomy and to address on-line uncertainty attributed from a robot itself or external environment, ideally we need the feedback control of the robot in task space with much higher bandwidth than that of the accumulated uncertainty. Therefore, high-speed sensing and high-speed control based on high-speed feedback information should be realized. This kind of system has been developed decades ago such as the 1 ms sensor-motor fusion system as presented in [10]. However, from the viewpoint that easy integration with a commercial robot's black-box controller (or even to consider the compatibility with the Industrial 4.0 [11]) is also an important issue, there is still no practical framework that can effectively address these issues.

In this chapter, we present the dynamic compensation framework to improve the autonomy of industrial robots. The dynamic compensation concept [12–16] is implemented based on high-speed sensory feedback as well as a coarse-to-fine strategy inherited from the macro–micro method [17, 18]. It should be noted that the macro–micro concept had been proposed several decades ago with the aim of enhancing system bandwidth for rigid manipulators and suppressing bending vibrations for flexible manipulators and this is not the scope of this study. In order to show the effectiveness of the proposed framework, several application scenarios are also presented.

2. Methodology

In this section, dynamic compensation framework under a hierarchical intelligent architecture would be presented. For the issue of asymptotic convergence, an intuitive analysis based on a simplified model will then be introduced. System integration with an industrial robot under the proposed framework would be addressed.

2.1 Dynamic compensation framework

The proposed framework for improving autonomy of industrial robots is based on a coarse-to-fine strategy as shown in **Figure 1**. Intelligence of a system is considered to be made up of two parts: action-level intelligence for motion control and planning-level intelligence for motion and task planning. Action-level intelligence represents low-level layer of the intelligence architecture and is referred to adaptability for motion control without sacrificing motion speed and absolute accuracy simultaneously, and we consider it as the foundation in implementing high-level intelligence. Real-time adaptation to both system uncertainty and environmental uncertainty enables a robot to focus on implementing high-level intelligence.

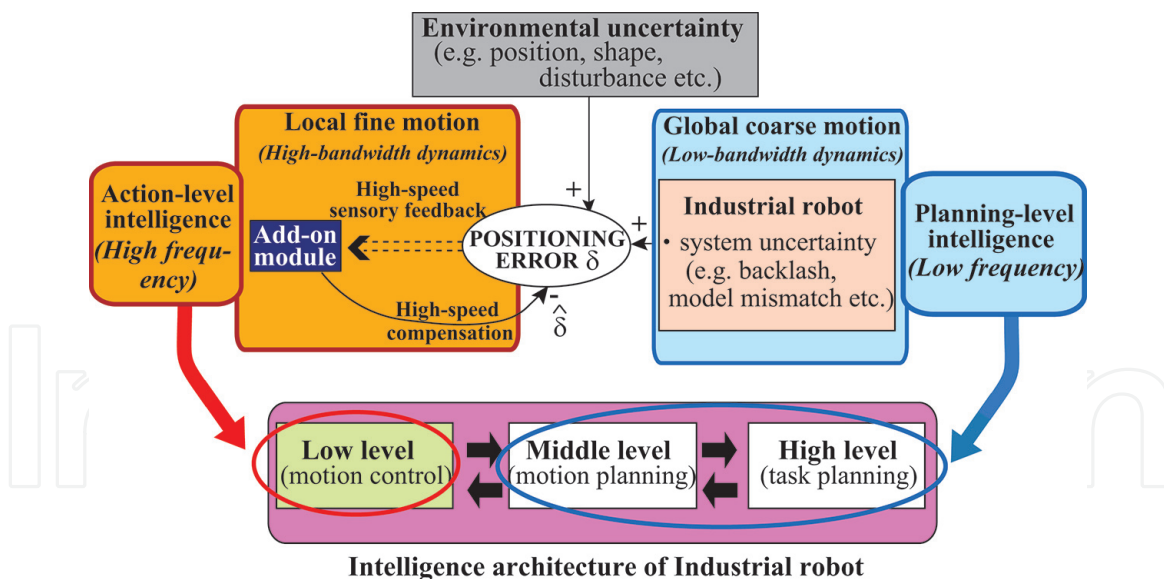


Figure 1.
Proposed framework.

Ideally, action-level intelligence is realized with high-frequency update rate to address the high-frequency part of on-line uncertainties and changes, whereas planning-level intelligence is allowed to implement with low-frequency update rate to tackle with the low-frequency part of uncertainties and changes. For implementation, traditional industrial robots are designated to conduct coarse global motion by focusing on planning-level intelligence. Concurrently, an add-on robotic module with high-speed actuators and high-speed sensory feedback is controlled to realize fine local motion with the role of implementing action-level intelligence.

2.2 Analysis on asymptotic convergence with a simplified model

In order to grasp the basic idea of the proposed framework, intuitive analysis on asymptotic convergence with a simplified model is provided with the assumption that the entire system is regulated in image space [13, 14].

As shown in **Figure 2**, an arbitrary industrial robot is controlled from position C toward position A (for simplification, the target is assumed to be motionless) with visual feedback information from a global camera. A direct-driven add-on module with high-speed actuators and high-speed sensory feedback is configured to its end effector. Specifically, we take the example of high-speed vision sensing for the add-on module. Initially, tool point B is assumed to overlap with tool point C . We refer to image features of target A and those of the robot's tool as ξ_a and ξ_c , respectively, from visual feedback of the global camera, and error e for regulation is noted as

$$e = \xi_c - \xi_a \quad (1)$$

Noting that $\dot{\xi}_c$ can be divided into two parts, motion effects corresponding to the main robot and the compensation module,

$$\dot{\xi}_c = J_r \dot{\theta}_m + J_c \dot{\theta}_c \quad (2)$$

where $\dot{\theta}_m$, $\dot{\theta}_c$ represent joint velocity vectors of the main robot and the compensation module, respectively, and J_r and J_c are the Jacobians (mapping from joint space to image space) of the main robot and the compensation module, respectively. We assume that the compensation module is not activated ($\dot{\theta}_c = 0$) and stays

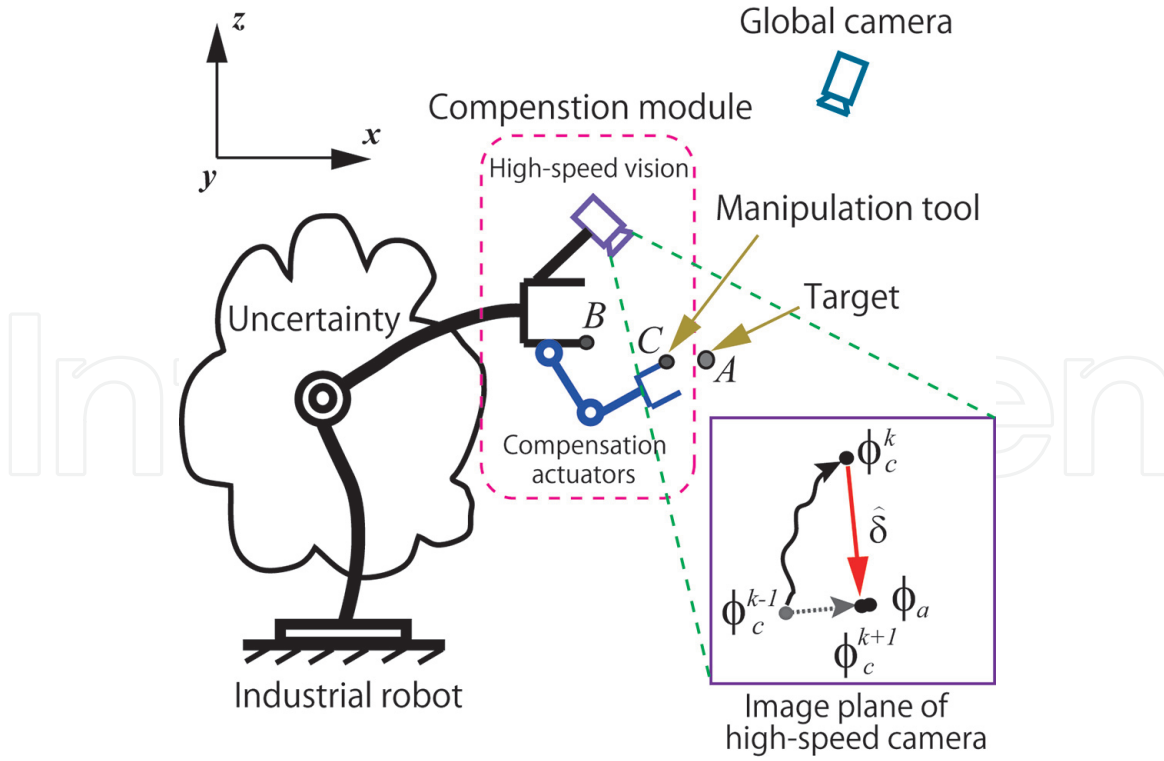


Figure 2.

A simplified model for addressing the dynamic compensation concept. Tool point C is controlled to align with target position A with the help of a compensation module, although the position of B is not certain due to the systematic uncertainty of the main robot. Image features of the global camera is represented by ξ , and image features of the high-speed camera is represented by ϕ .

still to retain the overlap between B and C. With the ideal case, exponential convergence of error regulation (for instance, ξ_c converges to ξ_a) can be obtained if we apply feedback control, such as

$$\dot{\theta}_m = -\omega J_r^+ e \quad (3)$$

where ω is a constant positive-definite coefficient matrix and J_r^+ represents the pseudo-inverse of J_r . However, in practice, the ideal visual-motor model of J_r^+ for an industrial robot is not available and is usually estimated with errors due to systematic uncertainties and inaccurate camera calibration. We denote the uncertain part as ΔJ_r^+ . The error dynamics with feedback control then become

$$\dot{\xi}_c = -\omega J_r (J_r^+ + \Delta J_r^+) e \quad (4)$$

and

$$\dot{e} = -\omega e - \delta \quad (5)$$

where δ represents the projected uncertainty in image space of the global camera. In this case, for instance, ϕ_c^{k-1} moves to ϕ_c^k rather than ϕ_a in image space of the high-speed vision system due to uncertainty, as shown in **Figure 2**. It should be pointed out that in spite of the uncertain term δ , the system is still assumed to conduct coarse positioning in the direction of the neighborhood of the target with the visual feedback of the global camera.

Now, let the compensation module be activated with motion such that $J_c \dot{\theta}_c = \dot{\phi}_c$. Let $\hat{\delta}$ represents the compensation module's motion observed by the global camera,

and let γ represents the conversion factor such that $\dot{\phi}_c = \gamma \dot{\xi}_c$. Note that γ can be perceived as time-invariant as the lens systems of both cameras are fixed and the relative position between the two cameras can be assumed as constant for a short-time period. The closed-loop system then becomes

$$\begin{aligned}\dot{e} &= -\omega e + \hat{\delta} - \delta \\ &= -\omega e - \tilde{\delta}\end{aligned}\quad (6)$$

where $\tilde{\delta} = \delta - \hat{\delta}$. In order to obtain the update law for $\hat{\delta}$, we choose the following Lyapunov function candidate

$$V(e, \delta) = e^T P e + \tilde{\delta}^T \Gamma^{-1} \tilde{\delta} \quad (7)$$

where P and Γ are two symmetric positive-definite matrices. Suppose that the direct-driven compensation module is feedback controlled by 1000 Hz of high-speed vision and the uncertain term δ due to the main robot which has big inertia can be approximated as a constant unknown term during the 1 ms feedback control cycle. Therefore, $\dot{\tilde{\delta}} = -\dot{\hat{\delta}}$, and the time derivative of V is given by

$$\begin{aligned}\dot{V} &= \dot{e}^T P e + e^T P \dot{e} - 2\tilde{\delta}^T \Gamma^{-1} \dot{\tilde{\delta}} \\ &= -2\omega e^T P e - 2\tilde{\delta}^T \Gamma^{-1} (\Gamma P e + \dot{\hat{\delta}})\end{aligned}\quad (8)$$

Apparently, $\dot{V} = -2\omega e^T P e \leq 0$ if we choose the update law for $\hat{\delta}$ as

$$\dot{\hat{\delta}} = -\Gamma P e \quad (9)$$

And resultantly, the control law for the compensation module should satisfy the following condition:

$$J_c \dot{\theta}_c = -\gamma^{-1} \Gamma P e \quad (10)$$

From the analysis above, we claim that asymptotic convergence is achievable using the proposed dynamic compensation in spite of systematic uncertainty in the main robot. Several issues should be noted here. First, the same conclusion can be drawn no matter how the main robot is controlled (in task space, as here, or joint space), and compensation capability can be further enhanced due to the fact that the control frequency of most commercial industrial robots is smaller than the 1000 Hz feedback control of the directly driven compensation module. Second, although dynamics are not fully incorporated within the analysis, our claim is still reasonable under the condition that the compensation actuator has a different bandwidth from that of the main robot. Third, although we have assumed the target to be motionless above, it is reasonable to apply the same analysis to cases where the target is moving but its motion is negligible in the context of 1000 Hz high-speed vision sensing. Moreover, although several robust and adaptive control approaches have been proposed (e.g., [3]) for direct control of robots with uncertain kinematics and dynamics, we note the advantages of our method in following two aspects:

1. The method here decouples the direct-driven compensation module and the main industrial robot and requires no changes to the main robot's controller.

On the contrary, traditional adaptive control methods need to directly assess the inner loop of a robot's controller (mostly not open), which is usually considered difficult both technically and practically.

2. It is difficult for traditional adaptive control methods to realize high-speed and accurate adaptive regulation due to the main robot's large inertia and complex nonlinear dynamics. With the philosophy of motion decoupling as well as adopting high-speed vision to sense the accumulated uncertainties, the proposed method here enables a poor-accuracy industrial robot to realize high-speed and accurate position regulation by incorporating a ready-to-use add-on module.

To summarize, the proposed dynamic compensation involves three important features:

1. The compensation module should be controlled accurately and sufficiently fast. Ideally, it has a much larger bandwidth than that of the main robot.
2. The sensory feedback for the compensation module should be sufficiently fast in order to satisfy the assumption $\dot{\delta} = -\dot{\delta}$.
3. The error value e is the relative information between the robot's tool point and the target in image coordinates, which can be observed directly.

Finally, it should be noted that since the add-on module works independently of the main robot's controller, optimal control of the system is another issue to address and is beyond the scope of this chapter.

2.3 Integration with conventional industrial robots

The proposed framework can be easily integrated with existing industrial robots. Usually, inner control loops of conventional industrial robots are black-box to end users due to issues concerning safety and intellectual property, and limited functions such as trajectory planning are available to users through interfaces provided by robot makers. In other words, it is difficult to incorporate external sensory information into the robot's inner control loop for motion control. On the other hand, as shown in **Figure 3**, control scheme for an arbitrary industrial robot and the add-on compensation module is separated with the proposed framework. Therefore, the compatibility of the proposed method is good as the industrial robot itself can be perceived as a black-box and users only need common interfaces for integration.

In the dynamic compensation framework, an industrial robot is designated for fast and coarse motion. Therefore, efforts for trajectory planning can be greatly reduced compared to traditional teaching-playback methods as well as adaptive control methods based on external sensory feedback. Coarse motion planning of the industrial robot can be either in a semiautonomous way or in an autonomous way. In the former method, teaching points covering a target trajectory can be very sparse as long as the target trajectory can be accessed by the add-on module. In the later method, coarse trajectory can be planned by utilizing external sensors (e.g., camera), and the calibration for sensor-motor mapping can be rough and easy. However, for conventional methods utilizing external sensors to realize accurate motion control of the industrial robot, calibration for sensor-motor mapping can be very complex and difficult.

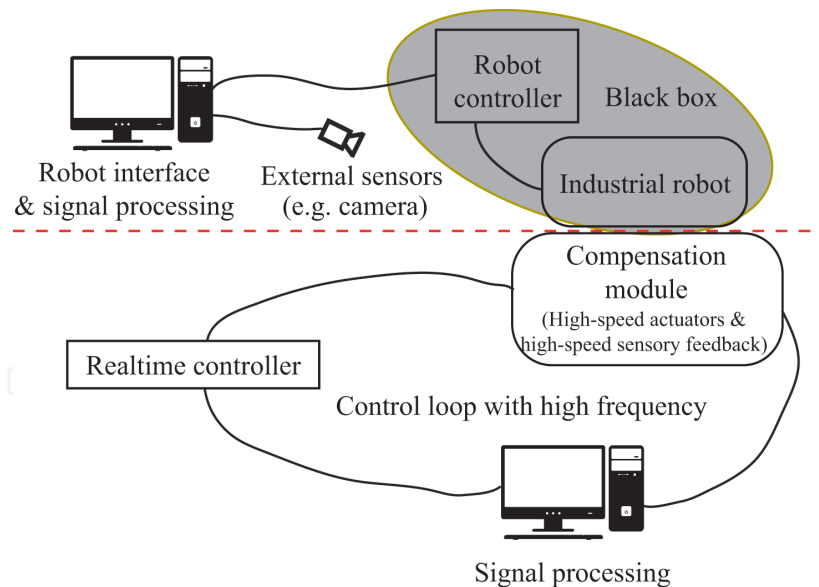


Figure 3.
Integration with conventional industrial robots.

Such as many other traditional methods, implementation of motion planning based on external sensory feedback involves two aspects: calibration for the mapping between the coordinates of sensory information and those of an industrial robot and signal processing to extract key points for motion planning from the sensory space according to a specified task. From the perspective of motion control, application tasks of industrial robots can be categorized into two kinds: set point motion control and trajectory motion control. For conventional methods, extracting key points and then realizing point-to-point motion control with accurate sensor-motor models for tasks involving set point motion (e.g., peg-in-hole task) is relatively easy to assure good accuracy. However, for tasks involving continuous trajectory control based on extracted key points and accurate sensor-motor models, good accuracy is still hard to realize due to the complex dynamics of an industrial robot during on-line moving. On the other hand, since an industrial robot is only asked to realize coarse motion in the proposed framework, calibration can be rough and easy, and motion errors due to the complex dynamics or even mismatched kinematics can be allowable, as long as the accumulated error is within the work range of the add-on compensation module. Application tasks with both set point motion control and trajectory motion control will be implemented with the proposed framework. Firstly, simplified peg-and-hole alignment in one dimension will be presented in Section 3. And then, contour tracing in two dimensions will be introduced in Section 4.

3. Application scenario 1: peg-and-hole alignment in one dimension

The simplified peg-and-hole alignment task was conducted to test the efficiency of the proposed method in realizing fast and accurate set point regulation under internal and external uncertainties [13]. The experimental testbed is shown in **Figure 4**. The add-on compensation module had one degree of freedom (DOF). A workpiece (metal plate with six randomly configured holes) was blindly placed on a desk for each experiment trial. The holes were 2 mm along the x -direction and were elongated in the y -direction to account for the fact that compensation was carried out only in the x -direction. A mechanical pencil with a diameter of 1.0 mm acting as the peg was attached to the linear compensation actuator, and the insertion

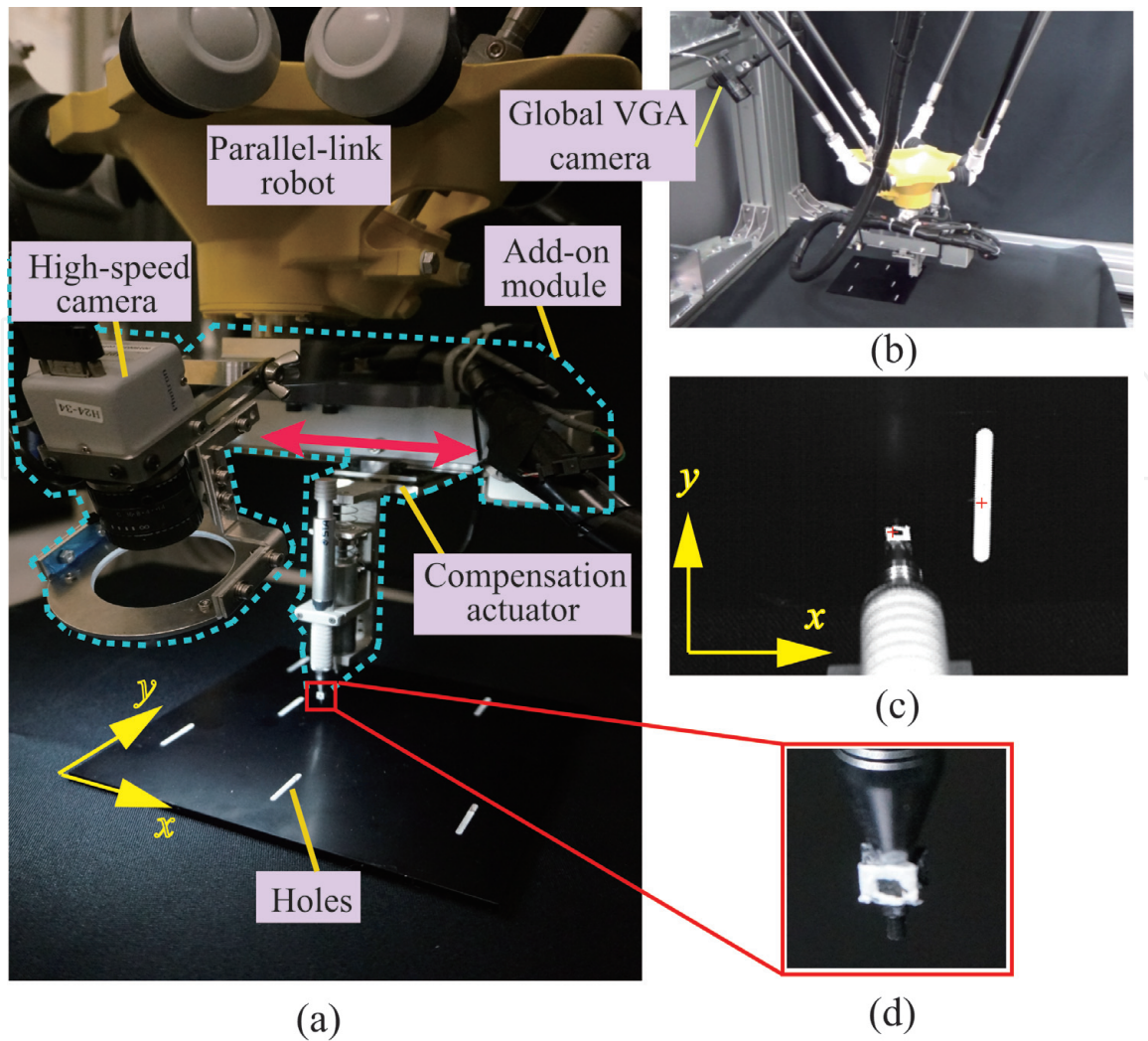


Figure 4. Experimental system [13]. (a) Overall setup: A one-DOF (x -direction) add-on module and a commercial parallel-link robot. (b) Global VGA camera for coarse motion planning of the parallel-link robot. (c) Detected marker and reference position (center of the nearest hole). (d) Marker representing the peg's position.

action was driven by an on-off solenoid. The insertion action was activated only if the error between the peg and the center of the hole in the x -direction was smaller than 0.8 pixels (corresponding to 0.112 mm) and lasted for more than 0.02 s. The insertion lasted for 0.3 s. We sought to insert the peg at the center of these holes. As can be seen from **Figure 4**, the holes formed the white parts of the otherwise black workpiece. Section 3.3.2 describes the process of detecting and obtaining the positions of these holes.

3.1 High-speed vision system with traditional configuration

As analyzed in Section 2.2, feedback information in task space for the add-on module should be high speed in order to satisfy the assumption $\dot{\delta} = -\dot{\delta}$. Here, high-speed vision system with traditional configuration (namely, imaging and image processing are conducted separately as shown in **Figure 5(a)**) was introduced. A Photron IDP-Express R2000 high-speed camera [19] (made by Photron, Japan) was used with the eye-in-hand configuration. The camera is capable of acquiring 8-bit monochrome or 24-bit color frames with the resolution 512×512 pixels at a frame rate of 2000 fps. It was connected to an image processing PC (OS, Windows 7 Professional; CPU, 24 core, 2.3 GHz Intel Xeon; Memory, 32 GB; GPU, NVIDIA

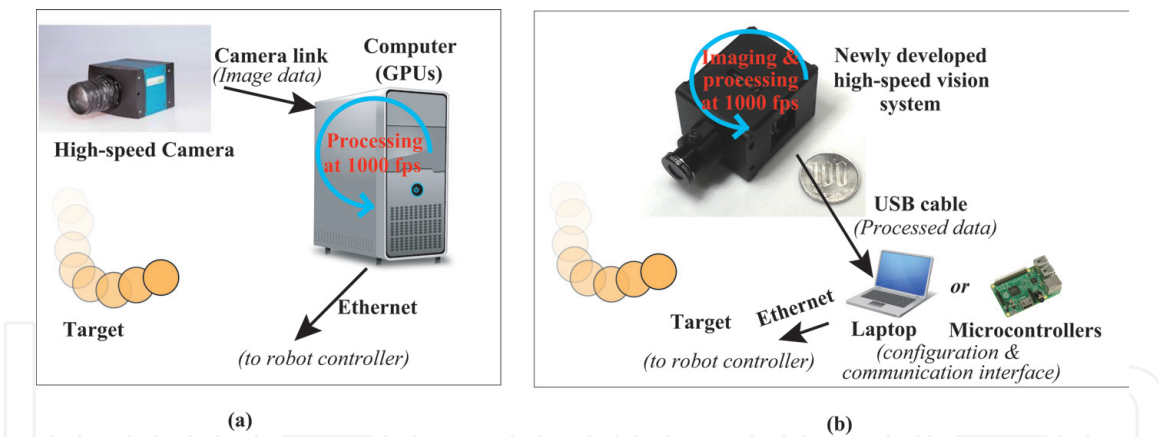


Figure 5. High-speed vision system [25]. (a) Traditional system with imaging and image processing conducted separately. (b) New high-speed vision system with high-speed imaging and processing integrated in one chip.

Stroke	Maximum velocity	Maximum acceleration	Weight
100 mm	1.6 m/s	200 m/s ²	0.86 kg

Table 1. Spec. of actuator for the one-DOF add-on module prototype.

Quadro K5200) (made by Dell Inc., USA), and the high-speed camera was configured with a working frame rate of 1000 fps.

Since control was limited to one dimension (along the x -direction in our case), the peg and the holes only needed to be aligned along the x -axis in the images. The peg was tracked using a marker fixed on the mechanical pencil at some distance from its tip (**Figure 3(b)**). In the captured images, it corresponded to roughly 9×9 pixel patches, and we employed a simple template-based search to find the location of the marker by minimizing the mean squared error. Following marker identification, we calculated its location at sub-pixel accuracy by computing image moments on the center patch. After locating the peg, we searched for the hole on a row in the image at a fixed distance from the detected marker in the y -direction. This image row was effectively binarized apart from the edge regions around the holes. We therefore searched for consecutive white regions (holes) and selected the one with center closest to the peg in the x -direction. The hole position was also computed with sub-pixel accuracy using image moments over the non-black region on the searched row. The processing ran within a millisecond using CUDA [20] to enable 1000 fps tracking of the positions of the pen and the hole. The high-speed camera is configured in such a way that the peg and each hole were both visible, and the relative error in image coordinates was sent to the real-time controller (**Figure 3**) by an Ethernet at a frequency of 1000 Hz. Since the high-speed camera was configured as the eye-in-hand, uncertainties due to the main robot as well as the external environment can be resultantly perceived as the variations of the hole's position, and they are accumulated within the relative error toward the peg.

3.2 Add-on compensation module with one DOF

The one-DOF add-on module with linear actuation was developed with specifications presented in **Table 1**. In accordance with the proposed dynamic compensation framework, the module was designed with large acceleration capability as well as being lightweight. The high-speed camera addressed above was configured with

a field of view of approximately 70 mm within the motion range of the linear actuator. Therefore, we had an approximate conversion of 1 pixel to be 0.14 mm.

As indicated in Section 2.2, control law of the compensation module should be developed according to Eq. (10). Obviously, Eq. (10) can be held with simple proportional derivative (PD) control or some other methods such as precompensation fuzzy logic control (PFLC) algorithm [21].

3.3 Coarse motion planning of industrial robot

A four-DOF parallel-link robot capable of high-speed motion was deployed to execute coarse global motion. A low-cost Video Graphics Array (VGA) camera (made by SONY, Japan) was mounted on the frame of the system and directed at the workspace. Using this camera, we fully automate the teaching task by detecting the rough position of holes in the main robot's coordinates.

3.3.1 Calibration issue

With the proposed framework, an exact calibration was unnecessary. Therefore, we did not worry about the intrinsic calibration of the VGA camera. Calibration simply involved computing the planar homography between the workpiece (a metal plate) in the main robot's coordinates and the image plane of the VGA camera. This was done by letting the main robot move to four points $\mathbf{X}_i = \{x_i, y_i, k_i\}^T$ directly above the workpiece and marking the corresponding locations $\mathbf{X}_{i'} = \{x_{i'}, y_{i'}, k_{i'}\}^T$ in the image. Here, the points \mathbf{X}_i and $\mathbf{X}_{i'}$ are homogeneous coordinates representing the 2D coordinates $\{x_i/k_i, y_i/k_i\}^T$ and $\{x_{i'}/k_{i'}, y_{i'}/k_{i'}\}^T$. The four points could be chosen randomly such that no three points were collinear. Using these four point correspondences $\mathbf{X}_i \leftrightarrow \mathbf{X}_{i'}$, we computed the homography $\mathbf{H} \in \mathfrak{R}^{3 \times 3}$ [22] by

$$\mathbf{X}_{i'} = \mathbf{H} \cdot \mathbf{X}_i (i = 1, \dots, 4). \quad (11)$$

Since we only needed a coarse homography, the procedure of choosing four point correspondences was rough and easy to implement. While we only consider one-dimensional compensation in this task, the same calibration procedure applies to full 3D compensation over a limited depth range so long as holes on the workpiece are observable by the high-speed camera for fine compensation. Hence, it would not be necessary to obtain 3D measurements using a stereo camera configuration or some similar mechanism. Since the camera was fixed in relation to the main robot's coordinate frame, the calibration procedure was implemented once and only needed to be performed again if the camera was moved or if the height of the workspace changed drastically.

3.3.2 Hole detection and motion planning

Usually, the model of holes on the workpiece should be known in order to detect them and calculate their locations. Here, we simplified the detection problem by utilizing the fact that the holes formed the white area of the black workpiece. The holes were identified and their locations were computed using image moments. The resulting points were transferred to the main robot's coordinate system using the homography computed in Section 3.3.1. Since the calibration mapping between the image and the robot's coordinates was rough, the detected points expressed in the robot's coordinates should reside in the neighbor area of their corresponding hole.

Following this, the shortest path connecting all of these points was calculated as a traveling salesman problem (TSP) [23]. Finally, the route (all points in order) was sent to the controller of the parallel-link robot to generate the corresponding motion, with the maximum motion speed set at 2000 mm/s.

3.4 Experimental result

The result of one experimental trial for continuous peg-and-hole alignment for six holes in the workpiece is shown in **Figure 6(a)**. The workpiece was placed randomly. **Figure 6(b)** shows the details of the second alignment. It can be seen that while the parallel-link robot executed coarse positioning at a high speed (maximum speed: 2000 mm/s), the hole's image position from high-speed vision in the x -direction did not become stable until after 2.1 s due to the fact that its rotational axis exhibited significant backlash. Nevertheless, the compensation module realized

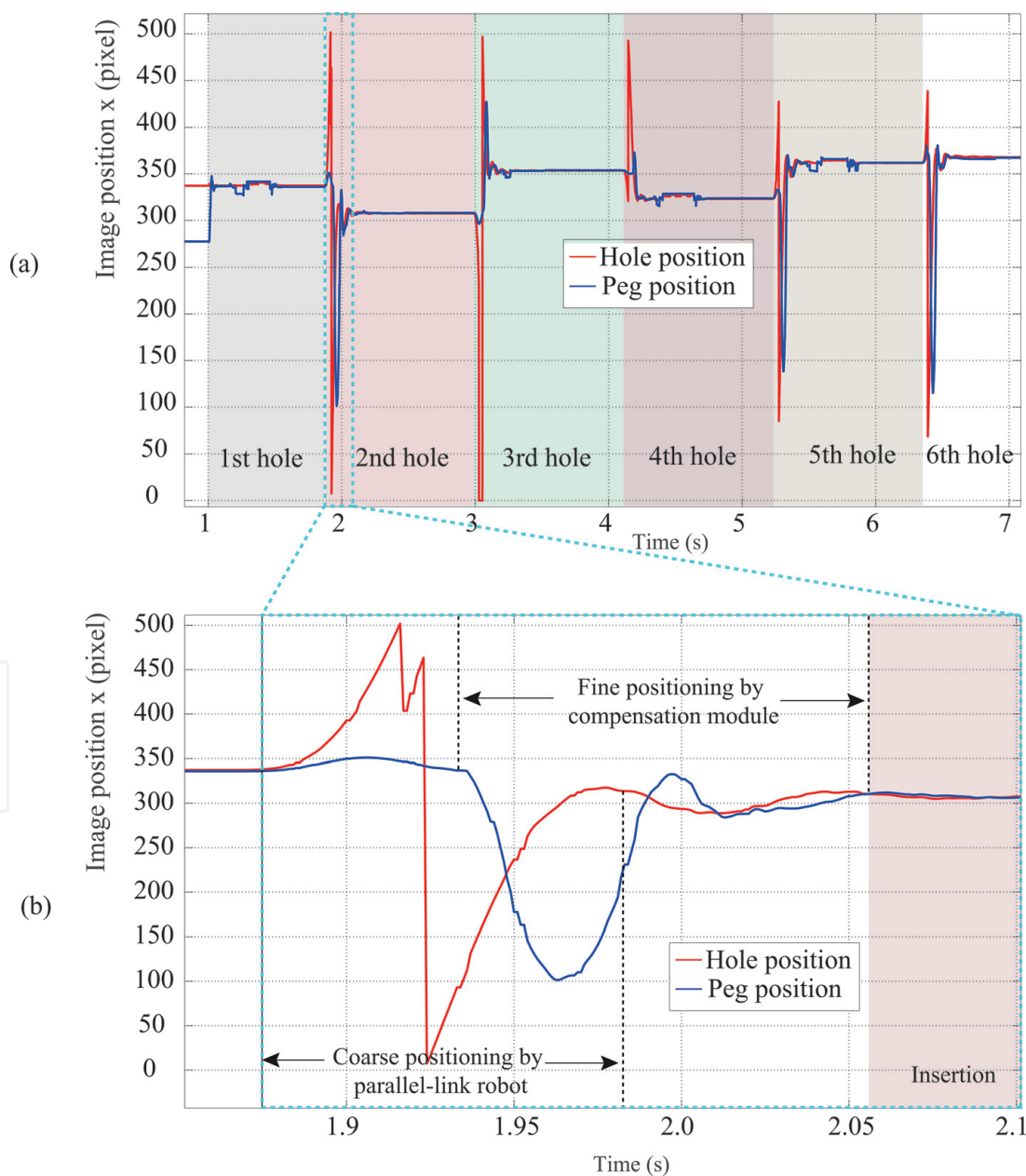


Figure 6. Image feature profiles of peg-and-hole alignment process [13]. (a) One experimental trial of continuous peg-and-hole alignment for six holes. (b) Zoomed details of the second alignment.

fine alignment within 0.2 s within an accuracy of 0.1 mm. For 20 trials with different positions of the workpiece, all alignments were satisfactory as the proper insertions were observed [13]. A video for the peg-and-hole alignment task can be found on the website [24].

4. Application scenario 2: contour tracing in two dimensions

The contour tracing task was conducted to verify the proposed method in realizing fast and accurate trajectory motion control under internal and external uncertainties [25]. Robotic contour tracing (contour following) is a useful technique in manufacturing tasks such as welding and sealing. Fast and accurate contour tracing under uncertainty from both a robot system and external environment is quite challenging. In this study, the task was confined in two dimensions. The experimental testbed is shown in **Figure 7**. The target is a closed curve with irregular contour pattern (2.5 mm width) printed on a paper which is placed on a stage. External disturbance is exerted on the stage to simulate environmental uncertainty. The same parallel-link robot was deployed as the main robot to execute the fast, coarse global motion. A two-DOF compensation module was configured at the end effector of the main robot. The same VGA camera was globally configured to realize the motion planning for the main robot's coarse positioning.

4.1 High-speed vision system with newly developed vision chip

Besides the method of realizing a high-speed vision system with traditional configuration as shown in **Figure 5(a)**, new high-speed vision system (**Figure 5(b)**)

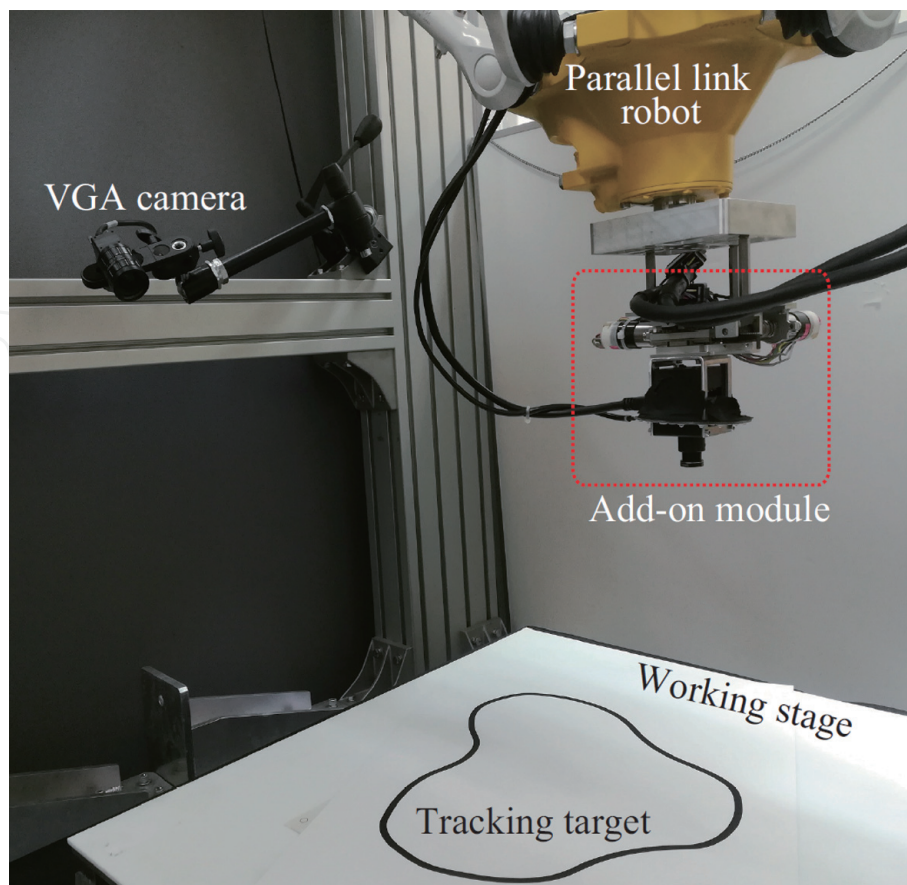


Figure 7.
Experimental setup for contour tracing task.

with newly developed vision chip is introduced. The new vision chip combines high-frame-rate imaging and highly parallel signal processing with high-resolution, high-sensitivity, low-power consumption [26]. The 1/3.2-inch 1.27 Mpixel 500 fps (0.31 Mpixel 1000 fps 2×2 binning) vision chip is fabricated with 3D-stacked column-parallel analog-to-digital converters (ADCs) and 140 giga-operations per second (GOPs) programmable single instruction multiple data (SIMD) column-parallel processing elements (PEs) for high-speed spatiotemporal image processing. The programmable PE can implement high-speed spatiotemporal filtering and enables imaging and various image processing such as target detection, recognition, and tracking on one chip. By realizing image processing on the chip, it can suppress power consumption to maximum 363 mW at 1000 fps. Comparing with conventional high-speed vision system, the new high-speed vision system will greatly save space and energy and is very suitable for compact usage in robotic applications. The high-speed vision was configured to work at 1000 fps with a resolution of 648×484 . Overall latency of high-speed visual feedback was measured to be within 3.0 ms [25].

4.2 Add-on compensation module with two DOFs

In order to accompany with the parallel-link robot to realize the two-dimensional contour tracing task, an add-on module prototype capable of realizing fine compensation in two dimensions was developed. The add-on module was with two orthogonal linear joints, and specifications for the actuators of the module were estimated by an accelerometer and are shown in **Table 2**. The total weight of the module was about 0.27 kg. The high-speed vision was configured on the moving table of the add-on module, and the tracing task was implemented in such a manner that the high-speed vision was guided to travel along the curve with the curve's center accurately aligned with the center (324,242) of the high-speed vision's images.

4.3 Coarse motion planning of industrial robot

The same as the peg-and-hole alignment task, the main robot's motion was planned using vision information from the globally configured VGA camera. The implementation involved exactly the same with the last task: a rough calibration and image processing to extract key points (via-points) of a target contour path considering the limited working range of the add-on module. Extraction of key points of a target contour was implemented in the following manner [25]:

1. As shown in **Figure 8(a)**, the image was binarized with a proper threshold, and a start point p_0 on the target contour was determined with the nearest distance to a user predefined point. p_c was initialized as p_0 .
2. A probing circle with its center at p_c was used to detect the intersection p_d with the target contour by predefined extraction direction.

Joint	Stroke	Max. velocity	Max. acceleration
x	20 mm	600 mm/s	63 m/s ²
y	20 mm	650 mm/s	70 m/s ²

Table 2.
Spec. of actuators for the two-DOF add-on module prototype.

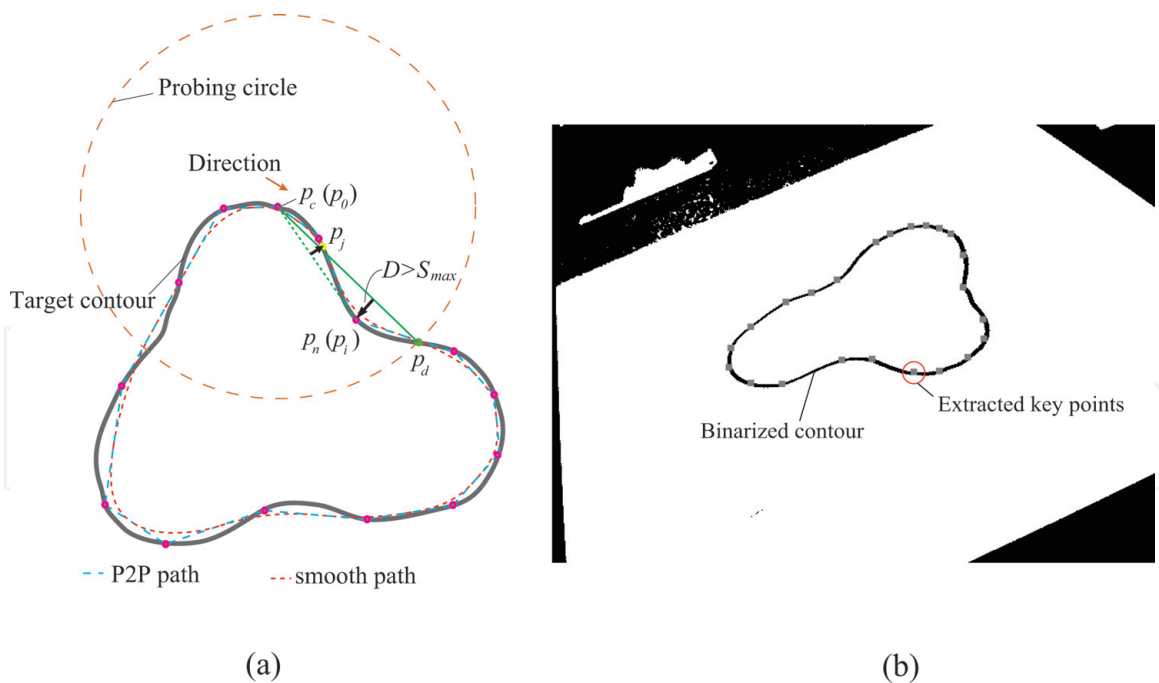


Figure 8. Coarse motion of the main robot. (a) Method for extraction of key points. (b) Extracted key points.

3. Point p_i starting from p_c along the extraction direction with a small step size δ was examined. If its distance to chord $\overline{p_c p_d}$ was within the work range (S_{max}) of the compensation module in image space, we then continue to examine the next point by increasing one step further. Otherwise, p_i was elected as the new extraction point p_n . With insertion of the new point p_n , points between p_c and p_n should be re-examined to see whether distance from p_j to chord $\overline{p_c p_n}$ was bigger than S_{max} or not. If it was true, another new extraction point at p_j should be inserted, and a recursive check for points between p_c and p_j should be conducted. Until all points between p_c and p_n were secured (the distance from each point to the corresponding chord was smaller than S_{max}), move the probing circle by updating p_c with p_n . And then the algorithm will return back to the last step until all the discretization points of the target contour were visited. The distance from p_i to chord $\overline{p_c p_d}$ was represented as D and was calculated by

$$D = \frac{|\overline{p_c p_i} \times \overline{p_c p_d}|}{|\overline{p_c p_d}|} \quad (12)$$

4. Points $[p_0, p_1, \dots, p_n]$ were the key points extracted from the target contour.

Usually, a commercial robot controller enables different methods of on-line path generation with selected key points. As an example, as shown in **Figure 8(a)**, a point-to-point (P2P) method that generated a path strictly passing through all key points with nonconstant velocity was included. On the other hand, a smooth path (100% smoothing factor) method would achieve a constant velocity profile while at the same time the exact generated trajectory would be not known to the user in advance. In many industrial applications, contour tracing with constant speed has significant advantages. It not only achieves good energy efficiency by reducing unnecessary acceleration and deceleration but also obtains better working performance in cases where work timing is critical, such as in welding. In this task, we

intended to perform contour tracing with a constant speed of the main robot. Therefore the smooth path (100%) method was adopted to control the main robot. However, this introduced additional source of uncertainty to the main robot's trajectory. One case of the extracted key points is shown in **Figure 8(b)**.

4.4 Experimental result

Before each experiment, the paper with irregular contour pattern (2.5 mm width) printed on it was placed randomly on the working stage. Key points (shown in **Figure 8(b)**) for coarse tracing motion of the parallel-link robot were extracted. The parallel-link robot was set to be 100% smooth motion with 400 mm/s speed. Along with the coarse tracing by the parallel-link robot, the compensation module was realizing fine local compensation in two dimensions with the visual feedback information from the new high-speed vision system.

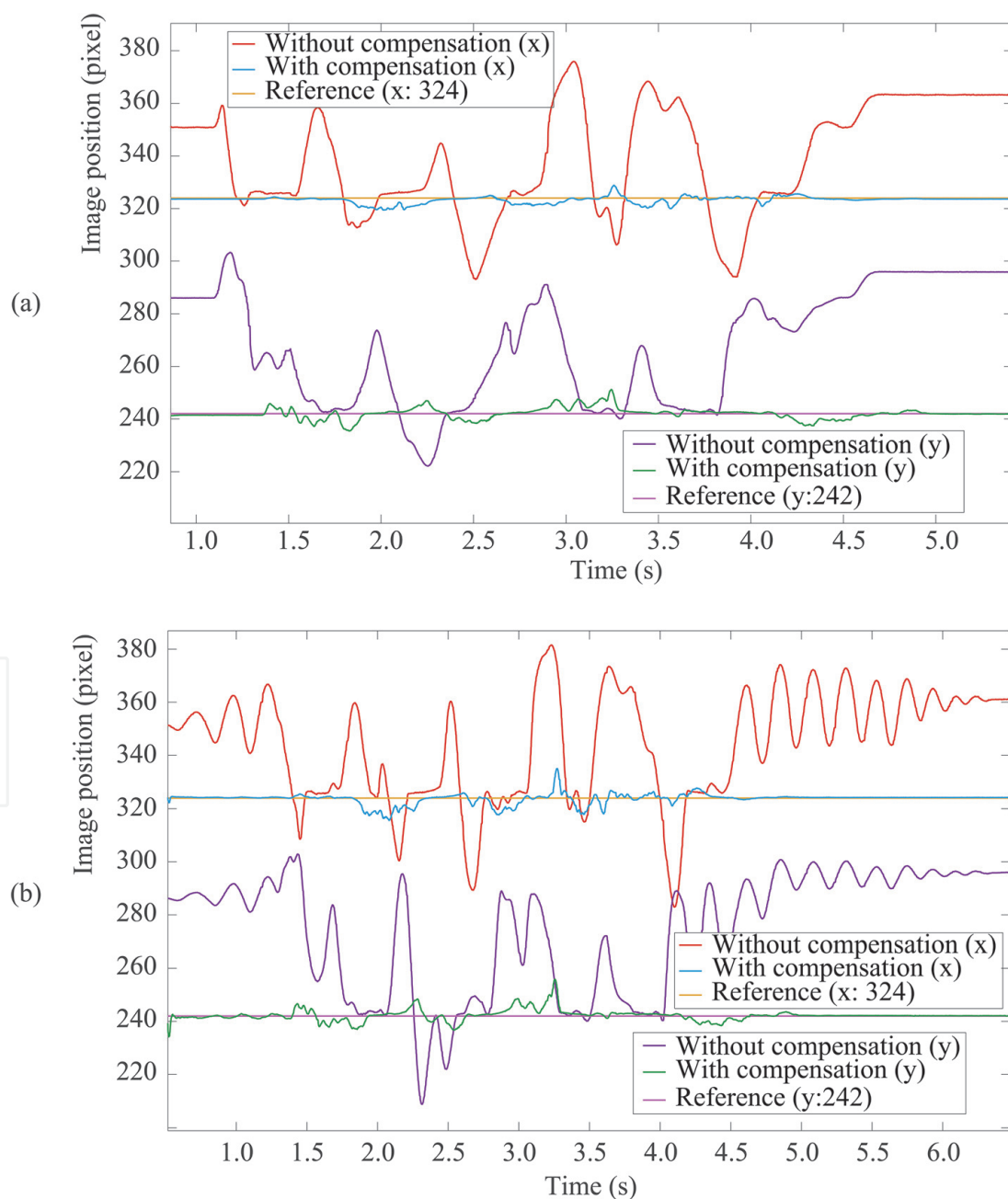


Figure 9. Result of contour tracing [25]. (a) Image profile of contour tracing without external disturbance. (b) Image profile of contour tracing with random external disturbance.

Tracing result of one trial without external disturbance is shown in **Figure 9(a)**. It was obvious that with motion planning by the global visual information, only coarse tracing by the parallel link was realized. With the cooperation between the parallel-link robot's global motion and the add-on module's local compensation, fine tracing with maximum error around 6 pixels (corresponding to 0.288 mm in the experimental setup) was realized [25].

In order to simulate external disturbance to the working target, the stage was manually vibrated in a random manner during the tracing process. The image profile of one trial is shown in **Figure 9(b)**. It demonstrated that the proposed dynamic compensation robot was capable of realizing accurate tracing even under disturbance from external environment, with the maximum tracing error around 9 pixels (corresponding to 0.432 mm) [25]. A video for the contour tracing task can be found on the website [27].

5. Discussion

5.1 For wider applications

The two application scenarios addressed above held the same fact that the add-on modules were directly utilized in task implementation and the payloads for the add-on module were small. As addressed in the analysis of the proposed framework, an add-on compensation module was assumed to have high-speed motion capability and ideally have a much larger bandwidth than that of the main robot. Therefore, payload for the add-on module should be very limited if the add-on module is directly utilized in task implementation. Then the readers may wonder whether the proposed framework is adoptable for wider applications where payloads for robots could be large. In these cases, we can actually utilize the dynamic compensation framework as a method of realizing autonomous teaching (programming) for an industrial robot. Let us take an example of a certain task with trajectory motion control in two dimensions where a manipulation tool (e.g., a torch for arc welding) would be too heavy for the add-on compensation module. The autonomous teaching procedure may implement as follows:

1. **Learning step.** The industrial robot with an add-on compensation module configured at its end effector realizes autonomous contour tracing (as addressed in Section 4) for a specific contour target (e.g., a seam for welding task). Configurations (in configuration space or joint space) of the industrial robot and the add-on module are memorized during the tracing process.
2. **Programming step.** Integrating the data of the add-on module into the configuration of industrial robot at each time point to generate the synthesized trajectory for the industrial robot.
3. **Manipulation step.** The industrial robot replaces the add-on module with the manipulation tool to its end effector and implements manipulation following the trajectory programmed in the last step.

It should be noted that the repeatability accuracy of an industrial robot is assumed to be good in these autonomous teaching applications. However, the proposed framework does have its limitations due to the fact that the on-line accumulated error should be observable by the high-speed sensory feedback in order to realize dynamic compensation.

5.2 Extensibility of the add-on module

Under the condition that the add-on module should have a much higher bandwidth than that of a companion main robot, the add-on module can be designed to be more than two DOFs. Specifically, it is suitable to adopt the parallel mechanism for an add-on module with more than three DOFs to assure motion accuracy. The high-speed sensory feedback for detection of on-line accumulated error can be high-speed vision with 2D/3D configuration or other forms of sensors depending on a specific task.

6. Conclusion

In order to improve the autonomy of industrial robots while at the same time keeping good accuracy under many uncertainties, we present a dynamic compensation framework under a hierarchical intelligent architecture based on a coarse-to-fine strategy. Traditional industrial robots are designated to conduct coarse global motion by focusing on planning-level intelligence. Concurrently, an add-on robotic module with high-speed actuators and high-speed sensory feedback is controlled to realize fine local motion with the role of implementing action-level intelligence. Main advantages of the proposed framework are summarized as follows:

1. Easy calibration for sensor-motor mapping as the industrial robot is designated for coarse motion
2. Good compatibility for industrial robots due to the fact that the control scheme for an arbitrary industrial robot and the add-on compensation module is separated
3. Good compatibility for artificial intelligence algorithms as the planning-level intelligence and action-level intelligence are decoupled

Author details


Shouren Huang^{1*}, Yuji Yamakawa² and Masatoshi Ishikawa¹

¹ Graduate School of Information Science and Technology, University of Tokyo, Tokyo, Japan

² Interfaculty Initiative in Information Studies, University of Tokyo, Tokyo, Japan

*Address all correspondence to: huang_shouren@ipc.i.u-tokyo.ac.jp

IntechOpen

© 2019 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Verdonck W, Swevers J. Improving the dynamic accuracy of industrial robots by trajectory pre-compensation. In: Proceedings of IEEE International Conference on Robotics and Automation. 2002. pp. 3423-3428
- [2] Maeda Y, Nakamura T. View-based teaching/playback for robotic manipulation. *ROBOMECH Journal*. 2015;2:1-12
- [3] Cheah C, Lee K, Kawamura S, Arimoto S. Asymptotic stability of robot control with approximate Jacobian matrix and its application to visual servoing. In: Proceedings of 39th IEEE Conference on Decision and Control. 2000. pp. 3939-3944
- [4] Cheah C, Hirano M, Kawamura S, Arimoto S. Approximate Jacobian control for robots with uncertain kinematics and dynamics. *IEEE Transactions on Robotics and Automation*. 2003;19(4):692-702
- [5] Zergeroglu E, Dawson D, Queiroz M, Setlur P. Robust visual-servo control of robot manipulators in the presence of uncertainty. *Journal of Field Robotics*. 2003;20(2):93-106
- [6] Piepmeier J, McMurray G, Lipkin H. Uncalibrated dynamic visual servoing. *IEEE Transactions on Robotics and Automation*. 2004;20(1):143-147
- [7] Wang H, Liu Y, Zhou D. Adaptive visual servoing using point and line features with an uncalibrated eye-in-hand camera. *IEEE Transactions on Robotics*. 2008;24(4):843-856
- [8] Bauchspiess A, Alfaro S, Dobrzanski L. Predictive sensor guided robotic manipulators in automated welding cells. *Journal of Materials Processing Technology*. 2001;109(1-2): 13-19
- [9] Lange F, Hirzinger G. Predictive visual tracking of lines by industrial robots. *International Journal of Robotics Research*. 2003;22(10-11):889-903
- [10] Namiki A, Nakabo Y, Ishii I, Ishikawa M. 1ms sensory-motor fusion system. *IEEE Transactions On Mechatronics*. 2000;5(3):244-252
- [11] Industry 4.0. Available from: https://en.wikipedia.org/wiki/Industry_4.0 [Accessed: May 4, 2016]
- [12] Huang S, Yamakawa Y, Senoo T, Ishikawa M. Dynamic compensation by fusing a high-speed actuator and high-speed visual feedback with its application to fast peg-and-hole alignment. *Advanced Robotics*. 2014; 28(9):613-624
- [13] Huang S, Bergström N, Yamakawa Y, Senoo T, Ishikawa M. Applying high-speed vision sensing to an industrial robot for high-performance position regulation under uncertainties. *Sensors*. 2016;16(8:1195): 1-15
- [14] Huang S, Yamakawa Y, Senoo T, Bergström N. Dynamic compensation based on high-speed vision and its application to industrial robots. *Journal of the Robotics Society of Japan*. 2017; 35(8):591-595
- [15] Huang S, Bergström N, Yamakawa Y, Senoo T, Ishikawa M. Robotic contour tracing with high-speed vision and force-torque sensing based on dynamic compensation scheme. In: Proceedings of 20th IFAC World Congress. 2017. pp. 4702-4708
- [16] Huang S, Bergström N, Yamakawa Y, Senoo T, Ishikawa M: High-performance robotic contour tracking based on the dynamic compensation concept. In: Proceedings

of IEEE International Conference on Robotics and Automation; 2016. pp.3886–3893

[17] Sharon A, Hogan N, Hardt E. The macro/micro manipulator: An improved architecture for robot control. *Robotics and Computer-Integrated Manufacturing*. 1993;**10**(3):209-222

[18] Lew J, Trudnowski D. Vibration control of a micro/macro-manipulator system. *IEEE Control Systems Magazine*. 1996;**16**(1):26-31

[19] Photron IDP-Express R2000 High-Speed Camera. Available from: <http://www.photron.co.jp/> [Accessed: May 4, 2016]

[20] CUDA Parallel Computing Platform. Available from: http://www.nvidia.com/object/cuda_home_new.html [Accessed: June 12, 2016]

[21] Huang S, Yamakawa Y, Senoo T, Ishikawa M. A pre-compensation fuzzy logic algorithm designed for the dynamic compensation robotic system. *International Journal of Advanced Robotics*. 2015;**12**, **12**(3):1

[22] Hartley R, Zisserman A. *Multiple View Geometry in Computer Vision*. Cambridge, UK: Cambridge University Press; 2004

[23] Lawler E, Lenstra J, Rinnooy K, Shmoys D. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Chichester, UK: John Wiley and Sons Ltd.; 1985

[24] Video for Peg-and-Hole Alignment Task. Available from: <http://www.k2.t.u-tokyo.ac.jp/fusion/pih/index-e.html> [Accessed: July 20, 2019]

[25] Huang S, Shinya K, Bergström N, Yamakawa Y, Yamazaki T, Ishikawa M. Dynamic compensation robot with a

new high-speed vision system for flexible manufacturing. *International Journal of Advanced Manufacturing Technology*. 2018;**95**(9–12):4523-4533

[26] Yamazaki T, Katayama H, Uehara S, Nose A, Kobayashi M, Shida S, et al. A 1ms high-speed vision chip with 3D-stacked 140GOPS column-parallel PEs for spatio-temporal image processing. In: *Proceedings of International Solid-State Circuits Conference*. 2017. pp. 82-83

[27] Video for Contour Tracing Task. Available from: <http://www.k2.t.u-tokyo.ac.jp/fusion/tracing/tracing.mp4> [Accessed: July 20, 2019]