Direct Motion Planning for Vision-Based Control

Roel Pieters, *Member, IEEE*, Zhenyu Ye, *Student Member, IEEE*, Pieter Jonker, *Member, IEEE*, and Henk Nijmeijer, *Fellow, IEEE*

***Abstract*—This paper presents direct methods for vision-based control for the application of industrial inkjet printing. In this, visual control is designed with a direct coupling between camera measurements and joint motion. Traditional visual servoing commonly has a slow visual update rate and needs an additional local joint controller to guarantee stability. By only using the product as reference and sampling with a high update rate, direct visual measurements are sufficient for controlled positioning. The proposed method is simpler and more reliable than standard motor encoders, despite the tight real-time constraints. This direct visual control method is experimentally verified with a 2D planar motion stage for micrometer positioning. To achieve accurate and fast motion, a balance is found between frame rate and image size. With a frame rate of 1600 fps and an image size of 160 $\times$ 100 pixels we show the effectiveness of the approach.**

***Note to Practitioners*— This method of visual control is motivated by the current state-of-the-art in display manufacturing. Traditional solutions for the manufacturing of displays assumes that the transformation between production head and control reference (e.g., kinematic or dynamic model) is known with high accuracy and that the display itself is rigid (i.e., fixed pitch between pixel centers). For flexible displays, this latter assumption is not the case. A method is proposed that takes these issues into account. A camera measures directly where the center of each display cell is located and generates online a trajectory for velocity motion control. This velocity reference is based on a smooth profile with a fixed velocity on cell centers and a higher velocity in between cell centers. This enables a higher overall velocity while ensuring a similar quality of printing compared to a constant velocity reference. Visual control and trajectory generation is executed at 1600 [Hz] with an image size of 160 $\times$ 100 pixels. Feedback is obtained only from visual input, the encoders present in the motors are *not* used.**

***Index Terms*—Inkjet printing, microrobotics, product as encoder, trajectory generation, vision-based control.**

Manuscript received November 27, 2013; revised July 01, 2014; accepted July 25, 2014. Date of publication August 13, 2014; date of current version October 02, 2014. This paper was recommended for publication by Associate Editor W. Sheng and Editor D. Tilbury upon evaluation of the reviewers' comments. This work was supported in part by Agentschap NL IOP Precision Technology–Fast Focus on Structures (FFOS) and in part by the Dutch Ministry of Economic Affairs (Pieken in de Delta)–Embedded Vision Architecture (EVA). This paper was presented in part at the International Conference on Advanced Concepts for Intelligent Vision Systems (ACIVS), Bordeaux, France, September 28–October 2, 2009, in part by the IEEE World Congress on Intelligent Control and Automation (WCICA), Jinan, China, July 4–9, 2010, and in part by the International Conference on Machine Vision Applications (MVA), Nara, Japan, June 13–15, 2011. *(Corresponding author: Roel Pieters.)*

R. Pieters is with the Multi-Scale Robotics Laboratory, Institute of Robotics and Intelligent System, ETH Zurich, CH-8092 Zürich, Switzerland (e-mail: r.s.pieters@ieee.org).

Z. Ye and H. Nijmeijer are with the Dynamics and Control Group, Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB, Eindhoven, The Netherlands (e-mail: z.ye@tue.nl; h.nijmeijer@tue.nl).

P. Jonker is with the Dynamics and Control Group, Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB, Eindhoven, The Netherlands, and also with the Delft Biorobotics Laboratory, Department of Biomechanical Engineering, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands (e-mail: p.p.jonker@tudelft.nl).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2014.2345954

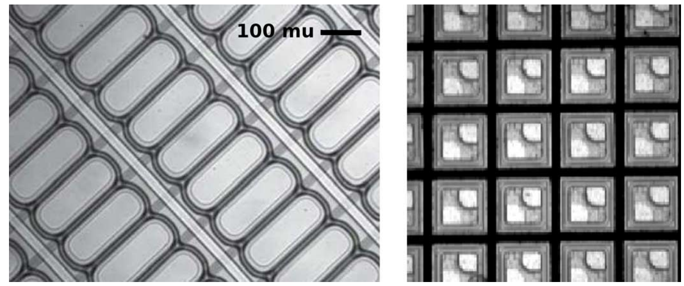

Fig. 1. Left: OLED substrate. Right: Transistors on a wafer.

I. Introduction

A. Motivation

The continuous consumer demand for better and faster electronics (e.g., displays for smart-phones, televisions and cameras) has led to the development of displays with increasingly higher resolution and increasingly smaller pixel size. The technology for manufacturing these display has to be improved or reinvented as well. In particular, current state-of-the-art display technology offers products which have a flexible or non rigid nature. The fabrication of these devices then becomes a clear challenge as the flexibilities cause high inaccuracies in the manufacturing process. For example, organic light emitting diode (OLED) displays require a printing task on every pixel cell, however, when the locations for printing are inaccurate or unknown (i.e., not measured directly), the display will not be manufactured correctly. These inaccuracies occur when the display is flexible, causing a mismatch between measurements of the display location (i.e., encoder-based) and the actual location of a display pixel. A different example is pick-and-place tasks of semiconductors on a wafer (see Fig. 1). The aim of this work is to develop a visual control system that circumvents these inaccuracies by taking a more direct approach towards sensing and control. In particular, the following developments are made:

- direct visual measurements;
- direct visual feedback;
- direct motion planning;

Direct visual measurements can determine accurately where a printing task should be executed. However, when vision becomes part of a control system, a number of problems may arise. Foremost, the fact that visual processing can take considerably more time to execute than a local control loop, demands the use of a double control loop structure. This is also known as indirect visual servoing. The local controller is executed at a fast rate (e.g., 1 [kHz]) to control the motion of the system, while a slow (e.g., 25 [Hz]) visual loop determines the motion of the system. This control structure is necessary to ensure stability and at the same time allow vision to be part of the control loop. When visual information can be fed back at an appropriate rate (i.e., such that motion is stable), a double control loop structure is no longer necessary and direct visual feedback can be achieved.

Furthermore, as visual servoing is a sensor-based control methodology, typical design of motion is executed on a path-planning level, where constraints are not directly taken into account. As the control structure now allows for direct measurements and feedback, and therefore trajectory tracking, the developed method also includes online motion planning with a constrained trajectory.

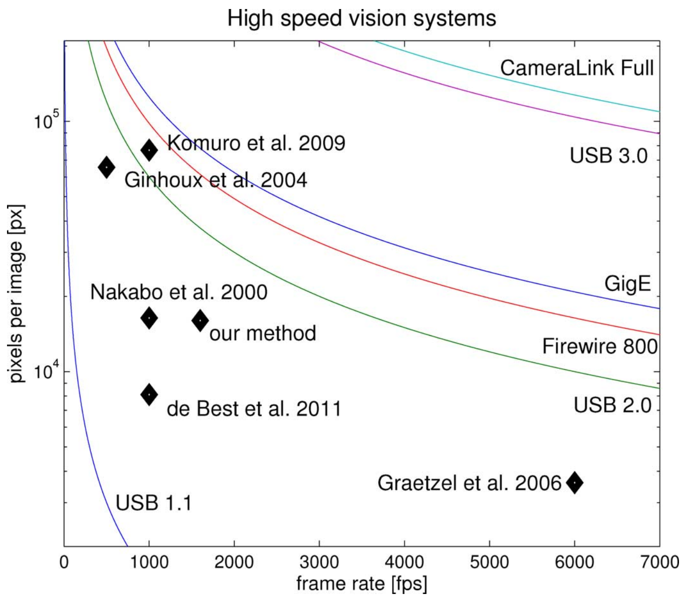

Fig. 2. Comparison of high-speed vision systems. The solid lines depict the theoretical limit of the communication protocol. The references are Komuro*et al.* [10], Ginhoux *et al.* [13], Nakabo *et al.* [9], Ogawa *et al.* [3], Graetzel *et al.* [12], and de Best *et al.* [14].

The main contributions of this work are therefore the integration, design and extension of traditional techniques (i.e., vision algorithms, motion planning methods) by using high-performance hardware (FPGA, camera). The developed system achieves a high control update rate (1.6 [kHz]) and is motivated for the application of industrial inkjet-printing. Moreover, in this context (i.e., inkjet printing), the method of closing the control loop with vision is fairly new. Typically, state-of-the-art inkjet printing solutions use visual feedback only for monitoring or evaluation.

B. Related Work

As a research field, visual servoing has matured greatly since it was first introduced in the 1980s [1]. The many forms in which visual feedback is used can be subdivided in a few groups: image-based or position-based visual servoing and eye-to-hand or eye-in-hand visual servoing. The former distinguishes the control in image space (IBVS) or cartesian space (PBVS), the latter describes the location of the camera; fixed in the real world and pointing towards the end-effector (eye-to-hand) or attached on the end-effector (eye-in-hand). Both IBVS and PBVS determine the error from image processing; either between image features (IBVS) or between object poses (PBVS). A survey and review can be found in [1] and [2]. Most attention in the past was on visual servoing for industrial robotic manipulators. More recently, much interest is also gained towards microscopic imaging systems with additional micro-positioning abilities. For example, Ogawa *et al.* propose a visual control system for tracking and directing motile cells using a high-speed tracking system [3]. Other, more traditional examples can be found in [4]–[7], and a short survey in [8].

Also, research in electronic systems has shifted towards visual control, where image retrieval and processing can be speeded up by using different processors (e.g., CPU, GPU, FPGA) or camera interfaces (GigE, CameraLink). An example of this is by Ishikawa *et al.* who have developed their own vision chip (128×128 [px] [9] and 320×240 [px] [10]) for processing at 1 [kHz].

Other high-speed vision systems, compared with respect to frame rate and image size can be found in Fig. 2, and a survey can be found in [11]. In this, Graetzel *et al.* [12] achieved a frame rate of 6 [kHz], however, with an image size of 60×60 [px]. In fact, visual control is not executed in this, as a static camera observes a fixed target (i.e., real-time wing beat analysis of drosophila). The method of Ginhoux *et al.* [13] shows a model predictive control scheme combined with visual servoing to track a beating heart in robotic surgery, which is achieved with a frame rate of 500 [Hz] and an image size of 256×256 [px]. From this comparison, it might be obvious that a high frame rate is chosen at the cost of a low resolution, which implies a lower accuracy as less pixels are available for analysis. This tradeoff is therefore the limiting factor in vision-based control systems. The two remaining methods, i.e., from de Best *et al.* [14] and our method, are discussed and presented in the next section.

In the context of inkjet printing, visual feedback is not commonly integrated. One example, however, can be found in [15], which uses a camera to monitor the properties of the resulting drop (i.e., vision does not close the control loop).

This paper is organized as follows. Section II presents the industrial application of inkjet printing and the high-speed visual servoing methodology. Section III presents the trajectory generation method and the visual control approach. Experimental results are presented in Sections IV, and Section V presents the conclusions.

II. Direct High-Speed Visual Servoing

As mentioned in the introduction, a basic visual servoing control architecture is divided into a slow visual reference loop (e.g., video rate) and a fast local joint control loop (e.g., kHz rate). Due to the slow visual update rate, the delay between a disturbance and a control action can be dozens of sample-times in joint control reference. However, as stated in [1], the availability of more and more computing power has enabled researchers to use vision for feedback at higher rates. If the camera is thus sampled fast enough (e.g., 1 [kHz]), this can reliably be used as single feedback for motion control. In the following section, our approach is motivated for repetitive patterns and the inkjet printing application.

A. Repetitive Product Pattern

The advantage of using the product directly for positioning can be explained best by a comparison. In traditional motion control, the motor encoder determines in part the overall system accuracy. This is true for systems in which the product is fixed with respect to the encoder and if the product itself cannot deform. If the location of the production head is relatively far from the motor encoder, vibrations can play a role in this large measurement loop, and the positioning accuracy can decrease. Similarly, if measurements are obtained to control the position of a motion stage, the transformation between production head and control reference (e.g., kinematic or dynamic model) has to be known with high accuracy. This knowledge is usually difficult to obtain and can even change over time. Furthermore, if measurements are done to control the position of a motion stage, the fixation of the product on the motion stage has to be rigid and identical for every new product, which implies a costly motion and fixation system. A direct, visual measurement system will effectively relate the position of the product with respect to the production head since these have the same coordinate frame. Moreover, when the product has a repetitive pattern, this can act repetitively as (visual) reference encoder and, if sampled fast enough, separate motor encoders become redundant.

Examples of repetitive patterns are for instance organic LED displays (OLED, Fig. 1) or semiconductors on a wafer substrate (Fig. 1). In both cases, a positioning task has to align the production head with respect to a repetitive pattern feature and perform a task. In the case of OLED manufacturing, this additional task consists of inkjet printing. In the case of semiconductor manufacturing, a pick-and-place task has to be carried out. Despite the difference in manufacturing, a similar approach towards using the product as encoder can be taken.

In particular, for the case of semiconductor manufacturing, de Best *et al.* [14] presented a visual control system with a frame rate of 1 [kHz]

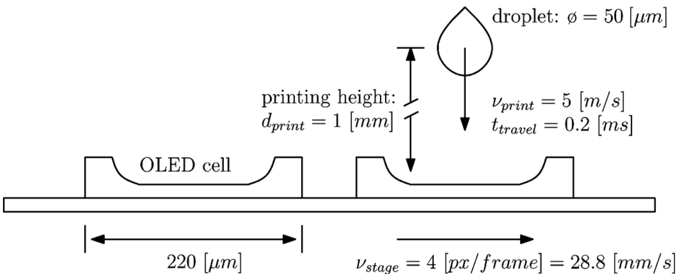

Fig. 3. Close-up of inkjet printing process. The travel time of the droplet causes a position error of 5.76 $[\mu\mathrm{m}]$. In order for the printing task to stay within the defined positioning tolerance of ± 10 $[\mu\mathrm{m}]$, the velocity of the motion stage at each OLED center should ensure $\nu_{\mathrm{stage}} = 4 \pm 2.9$ $[\mathrm{px/frame}]$.

and an image size of 90×90 [px]. In [14], results are shown where a 2D stop-and-go positioning task is executed with a positioning accuracy of ± 10 $[\mu\mathrm{m}]$ (3σ measurement variation: 0.3 $[\mu\mathrm{m}]$) and a delay of 2.5 [ms]. A repetitive pattern on a wafer consisting of semiconductor products (see Fig. 1) is used for direct feedback.

B. Industrial Inkjet Printing

The manufacturing of OLED displays requires an inkjet printing task on each individual OLED display pixel. As such, each pixel has to be aligned with the printing nozzle (print-head) and a printing action shoots a droplet of polymer into each cell (see Fig. 3 and [16]). For cost reasons, the manufacturing of such displays has to be done as fast as possible, implying also that the printing should be carried out as fast as possible. The obvious solution of printing in a stop-and-go manner, therefore, does not suffice. Instead, a printing task has to be executed on-the-fly, where the print-head moves with a fixed velocity over each OLED cell. If a display is a rigid structure (i.e., the pitch between OLED cells is equal) and the location of the display is known at all times, the printing task could be executed with a constant velocity and a constant drop-on-demand (DOD) print-frequency [17]. Existing research adopting this technique can be found in, e.g., [18] and [19], an overview of inkjet-based micro-manufacturing is given in [20]. However, due to the flexible nature of the display and the absence of a proper fixation system, a designed trajectory is a necessity. An additional reason for designing motion with a trajectory instead of a constant reference is the quality of the printing process. As the printing quality deteriorates with a higher velocity of the print-head with respect to the motion stage [21], a low velocity, when a printing action is executed, is desirable. In order to obtain a higher average velocity, a constant reference velocity should be avoided, and, instead, a designed trajectory should be employed.

Fig. 1 shows a microscopic view of an OLED display. The size of one OLED cell is 220×80 $[\mu\mathrm{m}]$, with a pitch in horizontal and vertical direction of 220 $[\mu\mathrm{m}]$ and 80 $[\mu\mathrm{m}]$, respectively. For the printing task, the print-head shoots a polymer droplet, which has a diameter of 50 $[\mu\mathrm{m}]$, at the center of each OLED cell (see Fig. 3 and [19]). In order to execute an accurate printing task, the delay of the printing task itself needs to be taken into account. The travel time of a droplet depends on the velocity of the droplet ν_{print} and the distance it has to travel d_{print}, and can be determined as

$$t_{\mathrm{travel}} = \frac{d_{\mathrm{print}}}{\nu_{\mathrm{print}}}. \tag{1}$$

Assuming a droplet print velocity of $\nu_{\mathrm{print}} = 5$ $[\mathrm{m/s}]$ and a printing height of $d_{print} = 1$ [mm], the travel time of a droplet equals $t_{\mathrm{travel}} = 0.2$ [ms]. Consider that a printing action is triggered when the motion stage is moving with a velocity of $\nu_{\mathrm{stage}} = 4$ $[\mathrm{px/frame}] = 28.8$ $[\mathrm{mm/s}]$ (for a 4.5 $[\mu\mathrm{m}]$ sized pixel and a frame rate of 1600 fps).

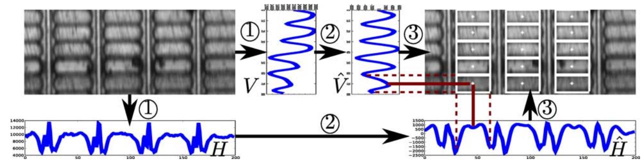

Fig. 4. Overview of the vision algorithm, which consists of three steps: ① projection, ② filtering, and ③ segmentation and image moment.

This implies that from the droplet leaving the nozzle to the droplet hitting the OLED cell, a distance of 5.76 $[\mu\mathrm{m}]$ or 1.28 [px] has been traveled by the motion stage. Similarly, the bounds on the velocity over the center of the OLED cell can then be determined. Assume that the position error for the printing process is tolerable at ± 10 $[\mu\mathrm{m}]$ from the center of the OLED cell. If the printing task is triggered exactly at the center of the OLED cell, the velocity which violates this error is then found as 6.9 [px/frame]. As such, the tolerance for the velocity of the motion stage at each OLED center is thus specified as: $\nu_{\mathrm{stage}} = 4 \pm 2.9$ $[\mathrm{px/frame}] = 28.8 \pm 20.9$ $[\mathrm{mm/s}]$.

As there already exists an error of 5.76 $[\mu\mathrm{m}]$ if the printing task is triggered exactly at the center of the OLED cell with perfect velocity tracking, a better solution is to predict when the print-head should be triggered. This is done by a linear predictor (i.e., α-β filter) which takes into account the delay due to the travel time of the droplet (i.e., $t_{\mathrm{travel}} = 0.2$ [ms]) as well as the delay between the trigger of the print-head and the droplet leaving the nozzle. This analysis assumes that the print-head is located at the center of the image, which might not be the case. For the actual printing task the delay due to this mismatch has to be taken into account as well.

III. Visual Control

Visual measurements are incorporated in a trajectory that is generated online, for every control cycle. The overall control structure is therefore defined as velocity trajectory tracking. These developments are explained in this section.

A. Image Processing

The vision algorithm takes a gray-scale image from the camera as an input, and performs three major steps to obtain the centers of the OLED cells, as illustrated in Fig. 4.

The first step is projecting the image horizontally and vertically, which results in two vectors H and V.

The second step is applying filters that eliminate illumination non-uniformity and noise. For the horizontal vector H, a moving-average filter is applied to obtained the background illumination $\bar{H}$. By subtracting $\bar{H}$ from H, the illumination-invariant vector $\tilde{H}$ is obtained. To eliminate the high-frequency noise, a low-pass filter G is applied to $\tilde{H}$ to produce a smooth vector $\hat{H}$. The vertical vector V is processed similarly to obtained $\hat{V}$. The third step is segmentation and image moment. The segmentation on $\hat{H}$ and $\hat{V}$ produces bounding boxes for OLED cells. Within each bounding box, the center of the OLED cell $\mathbf{p}_c = [p_{c,x},\ p_{c,y}]$ is obtained from the image moment.

B. Direct Trajectory Generation

The general idea of direct trajectory generation is that for each iteration a new motion profile for the next iteration is made, depending on the current constraints, the current state and the current final trajectory time. As the initial step, certain choices have to be made regarding the type of trajectory and its constraints.

The overall shape of the trajectory (and its time derivatives) as well as the degree of continuity $\mathcal{C}^{n_p}$ has to be specified. These choices can be easily incorporated in a polynomial trajectory by simply including or omitting constraint equations and by expanding or reducing the polynomial function. A trajectory of order n is therefore defined as

$$q(t) = a_0 + a_1 t + a_2 t^2 + \ldots + a_n t^n \tag{2}$$

with $t \in [t_i,\ t_f]$, where t_i indicates the initial time instant ($t = 0$) and t_f indicates the final time instant. A general solution is acquired by solving a system of linear equations

$$\mathbf{T}\mathbf{a} = \mathbf{x}_c \tag{3}$$

where $\mathbf{T}$ is the so-called Vandermonde matrix [22]

$$\begin{bmatrix} 1 & t_0 & & & \ldots & t_0^{n+4} \\ 1 & t_1 & & & \ldots & t_1^{n+4} \\ & & & & \vdots & \\ 1 & t_{n-1} & & & \ldots & t_{n-1}^{n+4} \\ 1 & t_n & & & \ldots & t_n^{n+4} \\ 0 & 1 & 2t_0 & & \ldots & (n+4)t_0^{n+3} \\ 0 & 0 & 2 & 6t_0 & \ldots & (n+4)(n+3)t_0^{n+2} \\ 0 & 1 & 2t_n & & \ldots & (n+4)t_n^{n+3} \\ 0 & 0 & 2 & 6t_n & \ldots & (n+4)(n+3)t_n^{n+2} \end{bmatrix}. \tag{4}$$

The unknown polynomial coefficients are $\mathbf{a} = [a_0,\ a_1, \ldots,\ a_n]^T$, and $\mathbf{x}_c$ lists the $(n+1)$ constraints that the polynomial should satisfy. The coefficients $\mathbf{a}$ can be computed as

$$\mathbf{a} = \mathbf{T}^{\dagger}\mathbf{x}_c \tag{5}$$

where $\mathbf{T}^{\dagger}$ represents the pseudo-inverse of $\mathbf{T}$. A more detailed explanation of this direct trajectory generation approach can be found in [23], a related method (i.e., online trajectory generation) in [24].

As mentioned in Section II-B, instead of a constant velocity trajectory, a smooth and varying trajectory should be employed for motion control. Such trajectory should be designed with a fixed velocity on OLED cells and a higher velocity in between OLED cells. This results in a higher average velocity for the manufacturing of a display.

A $\mathcal{C}^2$ continuous point-to-point trajectory is chosen, as this satisfies the requirements for a smooth and varying trajectory and does not excessively excite the motion system. This is due to the fact that acceleration discontinuities for adjoining trajectories are avoided. Moreover, choosing the trajectory as a fifth-order polynomial, implies that the sixth derivative is zero, which will minimize the integrated squared jerk [25].

The constraint vector $\mathbf{x}_c$ is obtained as

$$\begin{aligned} \mathbf{x}_c &= [\, x_i,\ x_f,\ \dot{x}_i,\ \ddot{x}_i,\ \dot{x}_f,\ \ddot{x}_f]^T \\ &= [\, x_{k-1},\ x_f,\ \nu_{k-1},\ \alpha_{k-1},\ \nu_{\text{drop}},\ \alpha_{\text{drop}}]^T \end{aligned} \tag{6}$$

in which image measurements are incorporated as $x_f = p_{c,x}$ for motion in x-direction. The motion of the print-head is designed such that at the center of each OLED cell the velocity and acceleration is equal to a predefined value

$$\begin{aligned} \dot{x}_f &= \nu_{\text{drop}} \\ \ddot{x}_f &= \alpha_{\text{drop}} \\ t_{\text{drop}} &= t_e \end{aligned} \tag{7}$$

in which, t_e is determined as

$$t_{e,l} = \max\left\{ \frac{15}{8}\frac{h}{v_{\max}},\ \sqrt{\frac{10\sqrt{3}}{3}\frac{h}{\alpha_{\max}}} \right\} \tag{8}$$

where $t_{e,l}, l \in \{v, \alpha\}$ is the execution time, $h = x_f - x_i$ and $v_{\max}$ and $\alpha_{\max}$ are the maximum velocity and acceleration, respectively [22]. This indicates that the motion is constrained by either a maximum velocity or acceleration and effectively enables a higher average velocity, while ensuring constraints on the center of the OLED cells.

The timing of each trajectory is thus determined by the visual measurements. This in effect implies that a trajectory is designed with respect to a global and a local kinematic constraint, i.e., arrival at predefined times with fixed local kinematic constraint

$$t_i = 0, \quad \text{and} \quad t_f = t_{\text{drop}} - \Delta t_{sum} \tag{9}$$

in which t_{drop} is taken from (7) and $\Delta t_{sum} = T_l n_k$ is the ascending trajectory time with T_l the local loop time with iteration count n_k.

The direct trajectory generator (DTG) generates a velocity trajectory to control the motion stage to move from one OLED cell to the next. For motion in x-direction, from initialization, the left most OLED cell (of the three horizontal cells in the field-of-view) is tracked and motion is generated to move it to the print-head position $p_{\text{print},x}$ with predefined constraints. When this position is crossed (i.e., before the next iteration), a new trajectory is generated. Similarly, the print-head should be triggered by the event of an OLED cell center crossing $\mathbf{p}_{\text{print}}$. However, as this crossing most likely occurs in between iterations, the exact trigger time t_{print} is predicted as

$$t_{\text{print}} = t_{k+1} + \Delta t_{cc} - t_{\text{travel}} - t_{\text{rem}} \tag{10}$$

$$\Delta t_{cc} = \frac{p_{\text{print},x} - \hat{p}_{c,x,k+1}}{\hat{p}_{c,x,k+2} - \hat{p}_{c,x,k+1}} T_l \tag{11}$$

where $t_{k+1} + \Delta t_{cc}$ is the estimate of the time until a cell center crossing, t_{travel} is the delay due to the travel time of the droplet through the air [see Section II-B and (1)] and t_{rem} is the remaining delay (e.g., due to data communication, position difference print-head and image center). This remaining delay can to be calibrated offline via the method presented in [21].

C. Velocity Trajectory Control

The motion of the xy-stage is velocity controlled due to the importance of a fixed velocity at each OLED cell center. This is necessary to guarantee a fast cycle time when manufacturing a display. The objective of the control system is therefore to track a time-varying reference trajectory $\dot{x}_d(t)$, in order to achieve

$$\|\dot{x}(t) - \dot{x}_d(t)\| \rightarrow 0 \text{ as } t \rightarrow \infty. \tag{12}$$

This control system is stabilized with a feedback compensator (i.e., a PID controller) and a feedforward compensator (i.e., mass and friction compensation). In discrete time, the velocity PID controller has the form (see also Fig. 5)

$$u_k = K_p \dot{e}_k + K_i \sum_0^T \dot{e}_k T_l + K_d \ddot{e}_k \tag{13}$$

where $\dot{e}_k = (e_k - e_{k-1})/T_l$, $\ddot{e}_k = (\dot{e}_k - \dot{e}_{k-1})/T_l$ and $e = x - x_d$, for motion in x-direction. Furthermore, K_p, K_i and K_d indicate respectively the proportional, integral and derivative gains. The feedforward compensation term for the mass and the friction of the motion stage is added as

$$F_{ff} = \hat{M}_t \ddot{x}_d + F_v \dot{x}_d + F_c sgn(\dot{x}_d) \tag{14}$$

where $\hat{M}_t$ is the estimated mass of the motion stage and where $F_v > 0$ denotes the viscous friction term, $F_c > 0$ denotes the Coulomb friction term and $sgn(\dot{x}_d)$ is the signum operator. This classical model of friction (see e.g., [26]) is sufficient to compensate for the major friction disturbance as occurs in the prescribed task (see Section IV).

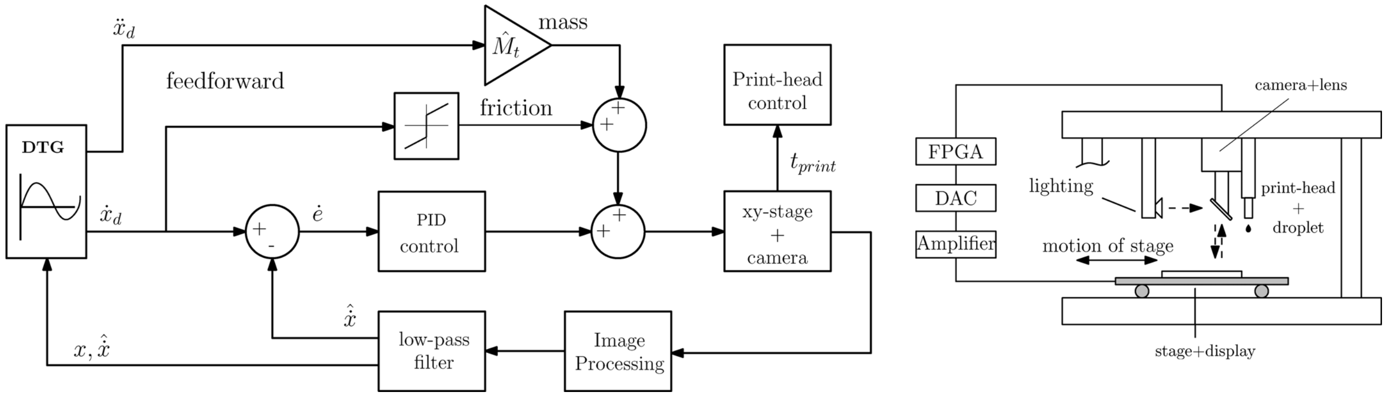

Fig. 5. Left: Control scheme. The DTG block generates a trajectory online based on image measurements. Trajectory tracking is achieved with a velocity PID controller, with an additional feedforward term to compensate for the mass and the friction of the motion stage. Right: Diagram of the visual control system for inkjet printing.

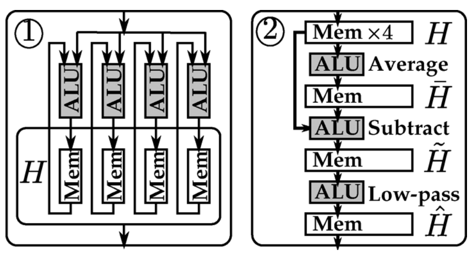

Fig. 6. Vision accelerator for ① projection and ② filtering, described in Section III-A. Only the processing of the horizontal vector H is illustrated. The processing of the vertical vector V is similar. (ALU: arithmetic logic unit. Mem: memory block.).

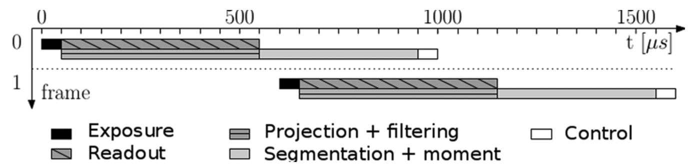

Fig. 7. Timing pipeline for the vision and control algorithm on FPGA. Processing starts directly when the first line of the image is received.

TABLE I
TIMING OF VISION PIPELINE

	Time
Start-to-end delay	1000 [μs]
Camera frame rate	1.6 [kHz]
Camera update	~600 [μs]
Exposure	50 [μs]
Readout	500 [μs]
Visual Processing	950 [μs]

IV. EXPERIMENTAL RESULTS

In this section the experimental setup is presented, and results are shown for the direct visual control methodology.

A. Experimental Setup

An experimental setup is developed which consists of two linear actuators (Dunkermotoren ServoTube STA11), a stationary camera (SVS-Vistek-340), and an FPGA (Xilinx Virtex-5 xc5vsx50t) for processing (see [14] and [27]). The camera sends monochrome images (8-bit per pixel) with a frame rate of 1600 fps and image size of 160×100 pixels directly to the FPGA via a CameraLink interface. Combined with a 1.5x magnifying lens (Opto-engineering MC1.50x) the images have a pixel size of 4.5 [μm].

As one pixel is represented by one byte, the effective network load for transferring the images at 1.6 kHz is roughly 26 MB/s (see also Fig. 2). As the proposed control method is a direct visual servoing approach, the control frequency is similar to the camera frame rate, i.e., 1600 Hz. It has to be mentioned that feedback is solely obtained from visual measurements, the local motor encoders which are present in the linear motion system are *not* used.

Visual processing is accelerated and optimized on a FPGA to utilize parallel processing as much as possible. The image sensor is directly connected to the processor such that processing starts directly when the first line of the image is received. The steps of high-computational complexity or high operations per data, and with regular memory access patterns, are mapped on to the dedicated accelerator on the FPGA (steps ① and ② as can be seen in Fig. 6).

The accelerator is configured to match the data rate of the camera (4 pixels per clock cycle), but can be tuned to handle a higher data rate if needed. The vision accelerator utilizes approximately 10% resource of the FPGA. Therefore, the accelerator can be scaled to support cameras with higher frame rates. More details of the vision pipeline on FPGA can be found in [28].

Fig. 7 and Table I show the timing breakdown of the complete image pipeline. It shows that the update rate is dominated (i.e., limited) by the transfer (readout) of image data to the processor.

B. OLED Cell Center Detection

As an initial step it is determined if a correction for lens distortion is necessary. The camera calibration method as presented in [29] is used due to the small field-of-view (i.e., ~ 6.2 [mm^2] for 640×480 [px]) and the limited depth of focus. Because of a low distortion lens, and the fact that the image for control only extends a maximum of 80 [px] from the center, the distortion has no significant influence.

Each 160×100 [px] image (i.e., $\sim$ 0.3 [mm^2]) contains 3×5 OLED cells. Fig. 8 shows a close-up of the result of the image processing steps as explained in Section III-A. The measurement noise has a standard deviation of $\sigma = 0.18$ [px] $= 0.85$ [μm]. As such, 99.7% of the measurements lie within the deviation of $3\sigma = 0.56$ [px] $= 2.55$ [μm], which is quite a substantial value considering the required accuracy of 10 [μm]. The ambiguity that can occur in the tracking of subsequent structures is avoided by limiting the velocity.

C. Trajectory Generation Results

Following, results are presented for visual trajectory tracking of a smooth point-to-point trajectory. Finally, a comparison with traditional encoder-based control is given.

Fig. 8. Output of center detection algorithm. Left: Image after segmentation. Right: Image depicting the detected OLED cells and their center.

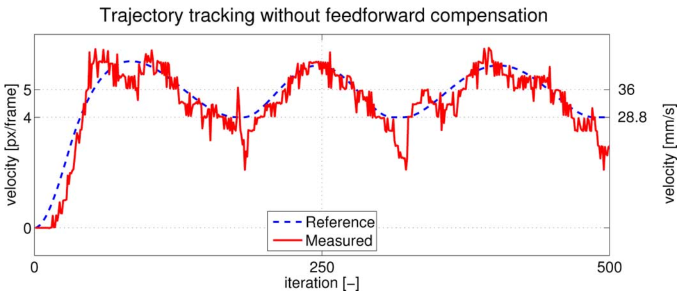

Fig. 9. Velocity trajectory control with DTG without compensation ($\dot{e}_{rms\_no\_FF} = 0.57\,[\text{px/frame}] = 2.56\,[\mu\text{m/frame}] = 4.1\,[\text{mm/s}]$). Especially at OLED cell centers (local minima where $\nu_{\text{drop}} = 4\ [\text{px/frame}] = 28.8\ [\text{mm/s}]$) only a PID controller proves not to be sufficient. Iteration 500 roughly compares to 0.31 [s].

D. Results for Point-to-Point Trajectory Tracking

To show the effectiveness of using a near-repetitive pattern for motion control the trajectory is designed as follows. From standstill a smooth velocity is designed to a fixed velocity (i.e., $\nu_{\text{drop}} = 4\ [\text{px/frame}] = 28.8\ [\text{mm/s}]$) and acceleration (i.e., $\alpha_{\text{drop}} = 0\ [\text{px/frame}^2]$) at an OLED cell center. The velocity in between the cell centers is chosen higher to obtain a higher printing throughput, and obtained by setting a maximum velocity for each velocity profile. This results in an average velocity for the trajectory of about $\bar{\nu} = 5\ [\text{px/frame}] = 36\ [\text{mm/s}]$, while for a constant velocity trajectory this would be equal to the velocity at the OLED cell centers, i.e., $\nu_{\text{drop}} = 4\ [\text{px/frame}] = 28.8\ [\text{mm/s}]$. This directly motivates the use of an online generated trajectory for motion control as a speed increase for printing of 25% is easily obtained. Depending on the limits of the actuators, this can be increased even more.

Fig. 9 presents the tracking results of the online generated trajectory with only a PID controller. It can be seen, at the start of the trajectory, that the static friction (stiction) takes several iterations to overcome. Furthermore, it shows that the viscous friction creates a delay between the reference velocity and the real (or estimated) velocity. This is particularly visible at relatively low ($< 4\ [\text{px/frame}] =< 28.8\ [\text{mm/s}]$) velocities. Furthermore, the friction of the system causes large disturbances at OLED cell centers (i.e., the local minima, where $\nu_{\text{drop}} = 4\ [\text{px/frame}] = 28.8\ [\text{mm/s}]$), and is most likely caused by the switching of sign of the acceleration (i.e., negative to positive). This delay and disturbance can be compensated for with a feedforward term which includes the mass of the motion stage as well as a friction compensation term as proposed in Section III-C. A final friction effect can be seen in the velocity range of $0 - 4\ [\text{px/frame}] = 0 - 28.8\ [\text{mm/s}]$ and reveals a stick-slip-like phenomena. This spontaneous jerking motion is caused by alternating sticking and sliding regimes in the lower velocity range.

Fig. 10 presents the tracking of the online generated trajectory with a PID controller and the mentioned compensation terms. The parameters for friction compensation are obtained via the method presented in [30] and via experimental tuning. In particular, the individual parameters of the Coulomb and viscous friction (i.e., F_c and F_v) are estimated based on open-loop measurements. A velocity ramp trajectory is executed as reference and from the resulting measurement response (i.e., velocity versus time) an initial estimate of the friction parameters can be retrieved. This initial guess is then tuned online (i.e., closed-loop)

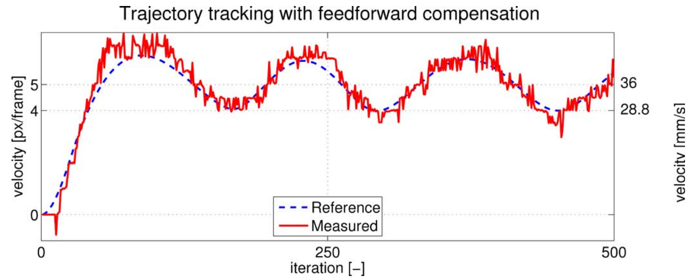

Fig. 10. Velocity trajectory control with DTG with feedforward compensation ($\dot{e}_{rms} = 0.40\,[\text{px/frame}] = 1.8\,[\mu\text{m/frame}] = 2.88\,[\text{mm/s}]$). The estimated velocity stays closer to the reference velocity compared to DTG without compensation. Iteration 500 roughly compares to 0.31 [s].

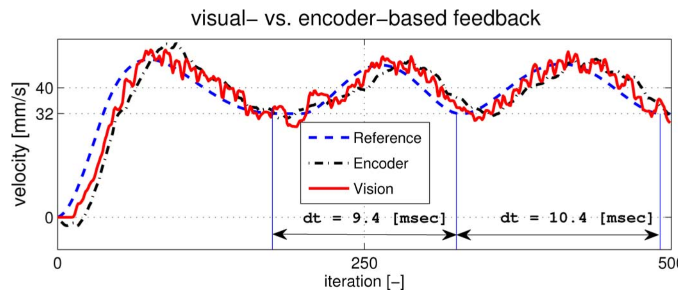

Fig. 11. Comparison of our proposed vision-based control approach and encoder-based control ($\dot{e}_{rms,vis} = 3.93\,[\text{mm/s}]$ and $\dot{e}_{rms,enc} = 5.79\,[\text{mm/s}]$, respectively). The timing between print actions is not constant (see vertical lines), and thus, an encoder-based approach is not suitable.

to obtain a better motion performance. The mass of the system $\hat{M}_t$ is estimated by weighing the motion system and tuned to obtain a decent performance. It can be seen that by compensation for the mass of the system as well as the viscous friction, the measured (or estimated) velocity follows the reference velocity more close. This is especially visible at relatively low ($< 4\ [\text{px/frame}] =< 28.8\ [\text{mm/s}]$) velocities. In the same velocity range, however, the stick-slip-like phenomenon is still visible. A compensation for this is not incorporated as the performance of motion control in this velocity range is not particularly of interest. This also holds for the stiction effect close to zero velocity.

The performance of trajectory tracking is evaluated by the root mean square (RMS) of the error velocity in Cartesian space. Without compensation of the friction and the mass of the system this is found as

$$\dot{e}_{rms\_no\_FF} = 0.57\left[\frac{\text{px}}{\text{frame}}\right] = 2.56\left[\frac{\mu\text{m}}{\text{frame}}\right] = 4.1\left[\frac{\text{mm}}{\text{s}}\right]$$

(see Fig. 9). When the compensation scheme is included the error RMS value is found as $\dot{e}_{rms} = 0.40\ [\text{px/frame}] = 1.8\ [\mu\text{m/frame}] = 2.88\ [\text{mm/s}]$, indicating a clear advantage of the compensation scheme (see Fig. 10).

A different important performance measure is the actual velocity on the center of the OLED cell. As can be seen in Fig. 9, there is a relatively large error between the reference velocity and the actual velocity on the OLED cell centers (i.e., local minima where $\nu_{\text{drop}} = 4\ [\text{px/frame}] = 28.8\ [\text{mm/s}]$), due to a poor controller. Fig. 10 shows that with a properly designed controller (i.e., including the feedfoward compensation) this error is clearly lower. Even though the velocity response has some delay, this amount of delay stays within bounds (i.e., $\pm 2.9\ [\text{px/frame}] = 20.9\ [\text{mm/s}]$ as determined in Section II-B) when considering the moment of printing: $\dot{e}_{rms} = 0.40\ [\text{px/frame}] = 2.88\ [\text{mm/s}]$.

E. Comparison With Encoder-Based Control

To further motivate our approach, we compare our method with traditional encoder-based motion control. For comparison, the reference velocity for both approaches is chosen as a smooth trajectory. Results can be seen in Fig. 11 and show that our method is equal or more accurate compared to encoder-based control ($\dot{e}_{rms,vis} = 3.93$ [mm/s] and $\dot{e}_{rms,enc} = 5.79$ [mm/s], respectively). This is because the built-in motor-encoder has a lower resolution by design and a higher amount of measurement noise. Moreover, it has to be mentioned that for traditional encoder-based control an online generated trajectory can *never* be determined, simply due to the lack of visual feedback. If visual information is not included, the inkjet printing task can never be successfully executed, as the location and time for printing can not be determined. This can be seen in Fig. 11 where the time between print actions is not constant, but determined at runtime.

A comparison to traditional visual control is intentionally omitted, due to the fact that these methods (i.e., position- or image-based visual servoing) minimize a position error and do not consider a velocity error, as is essential for the intended inkjet printing task.

V. Conclusion

This paper presented direct methods for vision-based control and the application of industrial inkjet printing. These methods are motivated by regarding the current state-of-the-art in visual motion control and industrial inkjet printing. For industrial inkjet printing (i.e., OLED display manufacturing by printing a droplet into each display pixel), the current state-of-the-art assumes that the pitch (or time) between individual printing actions is fixed and a constant print-frequency combined with motor-encoder feedback for control is sufficient for the manufacturing of displays. However, when this assumption no longer holds (i.e., a varying pitch due to a flexible display) current methods no longer suffice. This is mainly due to the fact that the product (i.e., the location for printing) is not directly measured. The proposed method takes this into account by designing a trajectory online based on direct visual measurements. The developments for this include a feature detection method for the detection of individual display cells (from a 160×100 [px] image) and the visual control method with direct, online trajectory generation (with update rate of 1600 [Hz]). As such, at each iteration, the next state of the trajectory is generated based on a predefined fifth-order point-to-point polynomial trajectory with predefined (i.e., $4\ [\mathrm{px/frame}] = 28.8\ [\mathrm{mm/s}]$) velocity on OLED cells, and a higher velocity in between OLED cells. This allows for a higher average velocity for the overall motion, which would be impossible for a constant velocity trajectory if a similar quality of printing should be ensured.

Acknowledgment

The authors would like to acknowledge all the work done by the Dynamics and Control Group, Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands.

References

[1] S. Hutchinson, G. Hager, and P. Corke, “A tutorial on visual servo control,” *IEEE Trans. Robot. Autom.*, vol. 12, no. 5, pp. 651–670, Oct. 1996.

[2] P. Corke, “Visual control of robot manipulators – A review,” in *Visual Servoing*. Singapore: World Scientific, 1993, pp. 1–31.

[3] N. Ogawa, H. Oku, K. Hashimoto, and M. Ishikawa, “Microrobotic visual control of motile cells using high-speed tracking system,” *IEEE Trans Robot.*, vol. 21, no. 4, pp. 704–712, Aug. 2005.

[4] B. Vikramaditya and B. Nelson, “Visually guided microassembly using optical microscopes and active vision techniques,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 1997, pp. 3172–3177.

[5] Q. Xu, Y. Li, and N. Xi, “Design, fabrication, and visual servo control of an xy parallel micromanipulator with piezo-actuation,” *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 4, pp. 710–719, Oct. 2009.

[6] Y. Anis, M. Holl, and D. Meldrum, “Automated selection and placement of single cells using vision-based feedback control,” *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 3, pp. 598–606, Jul. 2010.

[7] H. Chen and M. C. Shih, “Visual control of an automatic manipulation system by microscope and pneumatic actuator,” *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 1, pp. 215–218, Jan. 2013.

[8] P. Kallio, Q. Zhou, and H. Koivo, “Control issues in micromanipulation,” in *Proc. Int. Symp. on Micromechatronics Human Sci.*, 1998, pp. 135–141.

[9] Y. Nakabo, M. Ishikawa, H. Toyoda, and S. Mizuno, “1 ms column parallel vision system and its application of high speed target tracking,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2000, pp. 650–655.

[10] T. Komuro, A. Iwashita, and M. Ishikawa, “A QVGA-size pixel-parallel image processor for 1000-fps vision,” *IEEE Micro*, vol. 29, no. 6, pp. 58–67, Nov.–Dec. 2009.

[11] S. Kagami, “High-speed vision systems and projectors for real-time perception of the world,” in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recog. Workshops (CVPRW)*, 2010, pp. 100–107.

[12] C. Graetzel, S. Fry, and B. Nelson, “A 6000 Hz computer vision system for real-time wing beat analysis of drosophila,” in *Proc. IEEE/RAS-EMBS Int. Conf. Biomed. Robot. Biomechatronics (BioRob)*, 2006, pp. 278–283.

[13] R. Ginhoux, J. Gangloff, M. de Mathelin, L. Soler, M. Sanchez, and J. Marescaux, “Beating heart tracking in robotic surgery using 500 Hz visual servoing, model predictive control and an adaptive observer,” in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2004, pp. 274–279.

[14] J. de Best, R. van de Molengraft, and M. Steinbuch, “High speed visual motion control applied to products with repetitive structures,” *IEEE Trans. . Control Systems Technol.*, vol. 20, no. 6, pp. 1450–1460, Nov. 2012.

[15] M. Ezzeldin, P. van den Bosch, and S. Weiland, “Toward better printing quality for a drop-on-demand ink-jet printer: Improving performance by minimizing variations in drop properties,” *IEEE Control Syst. Mag.*, vol. 33, no. 1, pp. 42–60, Feb. 2013.

[16] S. Lee, J. Hwang, K. Kang, and H. Kang, “Fabrication of organic light emitting display using inkjet printing technology,” in *Proc. Int. Symp. Optomechatronic Technol.*, 2009, pp. 71–76.

[17] J. Zhou, J. Fuh, H. Loh, Y. Wong, Y. Ng, J. Gray, and S. Chua, “Characterization of drop-on-demand microdroplet printing,” *Int. J. Adv. Manuf. Technol.*, vol. 48, pp. 243–250, 2010.

[18] B. de Gans, P. Duineveld, and U. Schubert, “Inkjet printing of polymers: State of the art and future developments,” *Adv. Mater.*, vol. 16, no. 3, pp. 203–213, 2004.

[19] M. Ren, H. Gorter, J. Michels, and R. Andriessen, “Ink jet technology for large area organic light-emitting diode and organic photovoltaic applications,” *J. Imaging Sci. Technol.*, vol. 55, no. 4, pp. 1–6, 2011.

[20] , D. S. J. G. Korvink and P. J. Smith, Eds., *Overview of Inkjet-Based Micromanufacturing*. New York, NY, USA: Wiley, 2012.

[21] H. Wijshoff, “The dynamics of the piezo inkjet printhead operation,” *Phys. Rep.*, vol. 491, no. 4–5, pp. 77–177, 2010.

[22] L. Biagiotti and C. Melchiorri, *Trajectory Planning for Automatic Machines and Robots*. Berlin, Germany: Springer-Verlag, 2008.

[23] R. Pieters, A. Alvarez-Aguirre, P. Jonker, and H. Nijmeijer, “Direct trajectory generation for vision-based obstacle avoidance,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS), Workshop Robot Motion Planning: Online, Reactive, and in Real-Time*, 2012, pp. 109–114.

[24] T. Kröger and F. Wahl, “Online trajectory generation: Basic concepts for instantaneous reactions to unforeseen events,” *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 94–111, Feb. 2010.

[25] T. Flash and N. Hogan, “The coordination of arm movements: An experimentally confirmed mathematical model,” *J. Neuroscience*, vol. 5, no. 7, pp. 1688–1703, 1985.

[26] B. Armstrong-Hélouvry, P. Dupont, and C. Canudas de Wit, “A survey of models, analysis tools and compensation methods for the control of machines with friction,” *Automatica*, vol. 30, no. 7, pp. 1083–1138, 1994.

[27] Z. Ye, Y. He, R. Pieters, B. Mesman, H. Corporaal, and P. Jonker, “Demo: An embedded vision system for high frame rate visual servoing,” in *Proc. ACM/IEEE Internat. Conf. Distrib. Smart Cameras (ICDSC)*, 2011, pp. 1–2.

[28] Z. Ye, Y. He, R. Pieters, B. Mesman, H. Corporaal, and P. Jonker, “Bottlenecks and tradeoffs in high frame rate visual servoing : A case study,” in *Proc. IAPR Int. Conf. Machine Vision Appl. (MVA)*, 2011, pp. 55–58.

[29] R. Pieters, P. Jonker, and H. Nijmeijer, “Product pattern-based camera calibration for microrobotics,” in *Proc. IEEE Int. Conf. Image Vision Compu. New Zealand (IVCNZ)*, 2010, pp. 1–6.

[30] R. Kelly, J. Llamas, and R. Campa, “A measurement procedure for viscous and Coulomb friction,” *IEEE Trans. Instrum. Meas.*, vol. 49, no. 4, pp. 857–861, Aug. 2000.