

Received May 24, 2019, accepted July 1, 2019, date of publication July 4, 2019, date of current version July 22, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2926807

Eye-to-Hand Robotic Visual Tracking Based on Template Matching on FPGAs

ZHONG CHEN^{1,2}, SHUAI LI^{1,2}, NING ZHANG², YUYA HAO^{1,2}, AND XIANMIN ZHANG^{1,2}

¹Guangdong Provincial Key Laboratory of Precision Equipment and Manufacturing, South China University of Technology, Guangzhou 510641, China

²School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou 510641, China

Corresponding author: Zhong Chen (mezchen@scut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51875204, and in part by the Natural Science Foundation of Guangdong Province under Grant 2016A030313481.

ABSTRACT This paper presents a delta robotic visual-servoing tracking method using zero-mean normalized cross-correlation (ZNCC)-based grayscale template matching hardware core on field-programmable gate arrays (FPGAs). The concurrent FPGA-based ZNCC hardware core with cascading multiplication-accumulate (MAC) circuits is designed, which can largely reduce FPGA hardware resource consumptions. A compact optical imaging system with a front 45°-slant mirror and an optical filter film are proposed, which can efficiently filter out the background cluttered artifacts. The trajectory visual-servoing tracking and dynamic tracking experiments based on our built-up delta robotic visual tracking platform are implemented. The experimental results indicate that the presented FPGA-based embedded robotic visual tracking method can efficiently improve an object trajectory tracking performance.

INDEX TERMS Robotic visual tracking, template matching, FPGA.

I. INTRODUCTION

Visual tracking is to seek the solutions for continuously locating and following an object of interest in sequential image frames. Furthermore, visual servoing is to extend visual tracking into a control loop through visual feedbacks for regulating the machine's state automatically. Owing to its huge merits, visual tracking / servoing technology has been applied in many applications, such as surveillance, traffic regulation, autonomous vehicle navigation, micromanipulation, and industrial automation. Facing the lasting challenges from occlusion, background clutter, shape deformation, and scale variation etc., many visual tracking methods have been presented, such as area/feature matching-based method, classifier-based method, and particle filter etc. when considering the improvements at the different stage of visual tracking [1], [2]. Robotic visual servoing systems are mainly categorized into position-based visual servoing and image-based visual servoing [3]. In particular real-time and precision in position-based visual tracking should be paid more concerns in practical applications.

Matching-based visual tracking / servoing is more suitable for industrial scenarios, which is more regular than natural

The associate editor coordinating the review of this manuscript and approving it for publication was Bora Onat.

scenarios. Template matching based visual tracking is the most common approach owing to its simplicity on window-shifted matching computation, which kernel similarity measures include cross-correlation (CC), normalized cross-correlation (NCC), zero-mean normalized cross-correlation (ZNCC), the sum of squared differences (SSD), and the sum of absolute differences (SAD). Considering the limited performance of correlation filter tracker due to the direction of correlation computation and a single template with improper updating scheme, Liu presented a multi-template-matching-based tracker using mutual buddies similarity and memory filtering [4]. Li adopted SSD-based template matching through gradient-based inverse optimized fast searching to implement an in-plane micro motion tracker of the precision positioning stage based on micro-vision [5]. In order to improve tracking precision of visual servoing for microscopic objects, a six-axis motion visual tracker was presented, in which in-plane 3-DOF motion was obtained by NCC-based angle-discriminated template matching with affine constraints [6]. Feature matching based visual tracking / servoing commonly is more robust than template matching based methods. Liu presented a binocular micro-vision based 3D-motion tracker through high precision statistical estimation of feature points by Kalman filter [7]. A stereo micro-vision based optical fiber precision alignment system was built up,

in which the 5-DOF motion of the optical fibers was extracted by feature stereo correspondence matching for position-based visual servoing [8]. The shape features or angle features also were used in the visual servoing systems of mobile robot or an inverted pendulum [9], [10]. But due to its algorithm simplicity, template matching based trackers could attract the attention of many researchers more largely than feature matching based trackers if overcoming heavy computation consumptions from quantities of multiplication-accumulate (MAC) operations and exhaustive search.

In order to accelerate a template matching computation, especially ZNCC, the coarse-to-fine search [11], successive elimination technique [12], [13], integral image technique [14], [15], and bound partial correlation [16] soft-computing schemes were presented. Moreover, hardware-accelerated template matching methods, such as Graphics Processing Units (GPU), Digital Signal Processing (DSP) and Field Programmable Gate Array (FPGA), may be chosen due to their running-faster advantages than software-algorithm-accelerated methods. Therein, the FPGA technique is more suitable to be used to realize a template matching computation due to its intrinsic window-shifted operations in a way of in-stream and parallel processing. Because of the most straightforward implementation of the SAD algorithm, SAD based stereo matching/ tracking on FPGAs were found in many research works [17]–[19]. Although SAD-based FPGA circuit is simple, its object-locating precision heavily depends on the environmental lighting condition than other anti-illumination-variations algorithm. Local correspondence methods on FPGAs were successfully applied mostly in real-time stereo vision [19], [20] and seldom in visual tracking / servoing [21]. Some image-based registration techniques on FPGAs, such as Blob detection were adopted in robotic visual servoing [22]–[24]. But Blob-on-FPGA is a kind of binary algorithm, it hardly locates and tracks the objects precisely.

Based on the state-of-art of FPGA-based robotic visual tracking, we develop a ZNCC template matching based visual-servoing tracking system on a Delta robot in an eye-to-hand configuration according to real-time and precision requirements of robotic visual tracking. Because the captured images are usually background-cluttered, and the installation space for the imaging units is limited, we present a compact optical design that can aid in directly capturing a clear object-tracked image by optically filtering out background disturbs. Based on this built-up FPGA-based visual tracking system, various tracking experiments are implemented. This paper's main contributions are concluded as

1) A concurrent ZNCC template matching hardware core with cascading MACs circuits on FPGAs is designed for robot visual tracking, which consumes fewer FPGA resources than the other concurrent designs, and surpasses other FPGA-based template matching in precision and robustness.

2) In order to extend the field of view (FOV) as soon as possible and filter out the background clutter, a compact optical imaging system is designed, in which include a simple optical sheet filter and a 45°-slant plane mirror.

TABLE 1. Matching performance of ZNCC and NCC methods under uneven illumination.

Group No.	1		2	
Algorithm	NCC	ZNCC	NCC	ZNCC
Standard deviation(σ)	18.7	3.8	78.5	22.0

3) An eye-to-hand FPGA-based robotic visual-servoing tracking system is built up and verified experimentally in an actual scenario based on a Delta robot.

The following parts of this paper are arranged as follows. First, a concurrent ZNCC grayscale template matching method with cascading MAC circuits on FPGAs is presented in Section II. Then a compact optical imaging system with a front optical sheet filter and a 45°-slant plane mirror is designed, and robotic visual servoing scheme and ZNCC-based visual tracking core on FPGAs are described in Section III. In Section IV, the visual tracking experiments are implemented and discussed for the presented FPGA-based robotic tracking method. Finally, some conclusions are drawn in Section V.

II. ZNCC-BASED TEMPLATE MATCHING ON FPGAs

A. ZNCC-BASED TEMPLATE MATCHING

It is well known that the ZNCC algorithm has an obvious advantage of anti-illumination-variations compared to other correspondence matching methods, such as SAD, CC, and NCC, etc. In order to verify the anti-illumination-variation performance, we evaluate the matching performance between ZNCC and NCC methods under the simulated uneven illumination image on PC, which results are shown in Table 1 (The detailed experiment descriptions are omitted). Obviously, the anti-illumination-variation performance of ZNCC surpasses that of NCC.

But ZNCC algorithm needs large amounts of multiplication and addition operations, as may require lots of logic hardware resources in an FPGA. Meanwhile, ZNCC-based template matching runs in a mode of sliding window computation, as is also suitable to stream processing in FPGAs. This paper presents a concurrent ZNCC circuit with a cascading MAC scheme on an FPGA for less valuable resource consumption and strict timing. The grayscale template matching scheme is described in Fig. 1.

As shown in Fig. 1(a), the object boxed in red is found by template $T(x, y)$, as the object searching way is depicted in Fig.1 (b). After the template slides in the source image $I(x, y)$ from left to right and up to down at every step of one column pixels, the overlay pixels in the template image and reference image are used to compute similarity metrics referring to (1).

$$D(x, y) = \frac{\sum_{i=1}^M \sum_{j=1}^N (T_{i,j} - \bar{T}) \cdot (I_{i',j'} - \bar{I}_{x,y})}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N (T_{i,j} - \bar{T})^2 \cdot \sum_{i=1}^M \sum_{j=1}^N (I_{i',j'} - \bar{I}_{x,y})^2}} \quad (1)$$

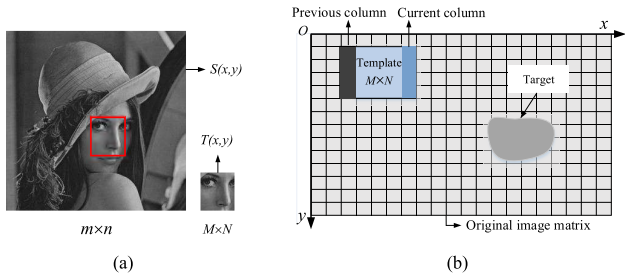


FIGURE 1. Working mechanism of grayscale template matching: (a) template $T(x, y)$, which size is $m \times n$ is searched in image $I(x, y)$ until maximum similarity of correspondence matching; (b) sliding window operations for similarity metrics computation between template matrix and original image matrix in the system clock cycle.

$$D(x, y) = \frac{C \cdot I \cdot II \cdot III}{\sqrt{IV \cdot C \cdot V \cdot (III)^2}}$$

$C = MN \sum_{i=1}^M \sum_{j=1}^N T_{i,j} \cdot I_{i',j'}$
 $I = \sum_{i=1}^M \sum_{j=1}^N T_{i,j}$
 $II = \sum_{i=1}^M \sum_{j=1}^N I_{i',j'}$
 $III = \sum_{i=1}^M \sum_{j=1}^N I_{i',j'}^2$
 $IV = \sqrt{MN \sum_{i=1}^M \sum_{j=1}^N T_{i,j}^2 - (\sum_{i=1}^M \sum_{j=1}^N T_{i,j})^2}$
 $V = \sqrt{MN \sum_{i=1}^M \sum_{j=1}^N I_{i',j'}^2 - (\sum_{i=1}^M \sum_{j=1}^N I_{i',j'})^2}$

FIGURE 2. Simplified ZNCC expression for in-stream and parallel computation on an FPGA, in which terminal I, II, III, IV, and V are the outputs of correlation computation between reference and template images, the related energy difference between reference and template, a sum of the reference image, and energy of the reference image, respectively.

where the pixel index $i' = x + i$, and the pixel index $j' = y + j$, $T_{i,j}$ is a pixel value of the template image $T(x, y)$ at i, j , \bar{T} is average of all pixel values in the image $T(x, y)$, $I_{i,j}$ is a cached image pixel value from the video stream at the template window index i, j , and $\bar{I}_{x,y}$ is average of all pixel values in the cached image.

Due to large amounts of MAC computations in (1), ZNCC formula should be simplified further considering stream processing and parallel computation on an FPGA. The simplified equation [25] is formulated and shown in Fig.2.

In the simplified ZNCC expression, C and output II/IV are constant and can be precomputed outside FPGAs. So these constants are integrated into the final FPGA circuit. Moreover, the output I/III/V should be computed in an FPGA due to different reference images. In the three subunits, correlation coefficient (CC) subunit needs large amounts of MAC operations. Because the template image should be feed into computation modular of unit I, it should be pre-stored in the Block-Ram in the FPGA. Some researchers proposed some solutions. Lindoso proposed a cascading MAC circuit with valuable DSPSlice resources in an FPGA for CC computation [26]. Zhou developed a reconfigurable hybrid MAC scheme combined cascading and concurrent MAC computation [27]. In contrast, we present concurrent computation architecture with cascading MAC computation for CC computation, which will be described in the next subsection. In our designed practical circuit, division and square

root operations are implemented by calling *LPM-DIVIDE* IP and *ALTSQRT* IP provided in IDE software Quartus II, respectively.

B. IMAGE CACHING ON FPGAs

According to the simplified ZNCC formula, the reference image from the camera and the template image should be cached before further processing in an FPGA. Comparing to a memory-based image caching method, a shift-register-based image caching method can easily satisfy real-time processing requirements for an imaging pixel stream due to its inherent concurrent characteristics. So we adopt a synchronized FIFO buffering technique for multi-row pixel buffering, which buffering mechanism is shown in Fig. 3.

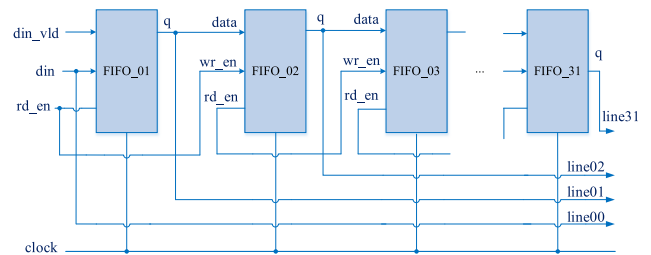


FIGURE 3. Hardware architecture of multi-row buffering using synchronized FIFO for 32-row buffering.

In Fig. 3, every FIFO shares the same clock signal *clock*. The read-enable terminal *rd_en* in the current FIFO is linked to the write-enable terminal *wr_en* in the next neighboring FIFO. So the output *q* can be output to the data input terminal *data* of the next neighboring FIFO synchronically. After finishing data pouring to every FIFO, every row grayscale pixels are buffered, meanwhile, every FIFO outputs its data-buffered signals *line01*, ..., or *line31*.

In order to understand the buffering process intuitively, timing sequence logic in the multi-row buffering is depicted in Fig. 4. It is obvious that all row data keep in alignment after finishing 32-row buffering, as indicated by a jumping signal *dout_vld* for the windowed pixels output valid. The data alignment also can be observed in the red box zone in Fig. 4. Obviously, after alignment, the serialized imaging pixel stream is projected to an image matrix for the following template matching.

In order to construct correlation operations in an FPGA, a FIFO-based column-scanning 32-row-pixel buffering scheme is designed, which can pipeline buffered sub-images for template matching according to a special row-column scanning mode, as depicted in Fig. 5. This buffering process is a kind of shift-register-based buffering method. It is obvious that template matching in a size of 32×32 runs once after 32 clock cycles. This kind of concurrent operation makes template matching run very fast, which is different from the memory-based approach [25]. But there is a shortcoming of low matching-based location precision at some position due to template size step in the column-scanning process along the row directions but high location precision due to

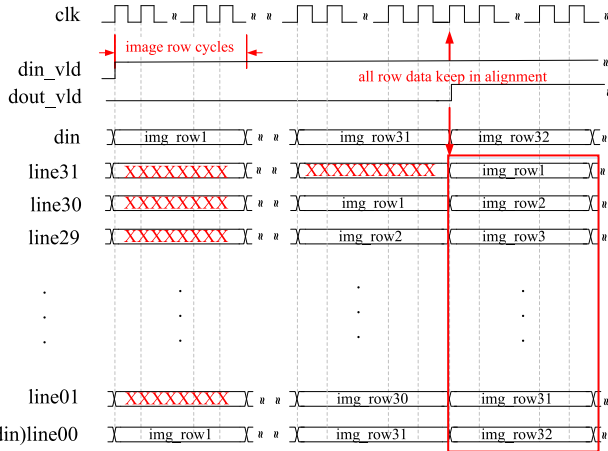


FIGURE 4. Timing sequence logic for 32-row FIFO buffering.

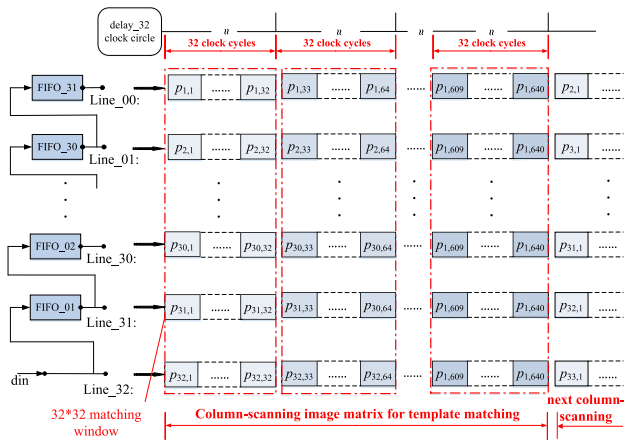


FIGURE 5. Concurrent pipelining buffering architecture for 32×32 windowed matrix pixels. $p_{i,j}$ represents an image pixel, which i and j are image pixel coordinates at rows and columns respectively.

one-pixel step along the column direction. But high matching-based location precision can be obtained for motion objection tracking when adopting a scheme of keeping high correlation results and neglecting low correlation results. This paper adopts this kind of approach for high precision real-time object tracking.

C. FPGA-BASED ZNCC DESIGN

According to the above simplified ZNCC expression, concurrent pipelining hardware architecture is designed as shown in Fig. 6. The REGs in Fig. 6 represent the registers which function as the role of data buffering in the pipelining processing at every clock cycle.

In order to realize concurrent operation and resource reuse in an FPGA, the hardware architectures for the unit I, III, and V in the ZNCC expression are designed as shown in the following. The circuit architecture for unit III in Fig. 2 is depicted in Fig. 7, in which inputs come from the outputs of a buffering circuit in Fig. 3, and functions as integration on the windowed camera pixel stream.

The circuit architecture for unit V in Fig. 2 is depicted in Fig. 8, in which two-row buffering pixels of the camera

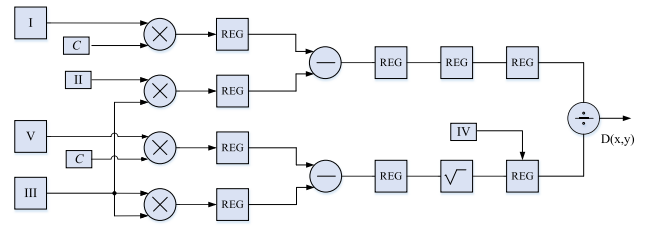


FIGURE 6. Pipelining architecture of ZNCC circuit, in which REG is a register to store the results at each cycle.

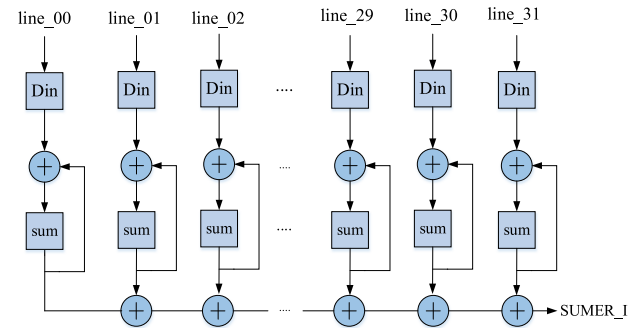


FIGURE 7. Circuit architecture for a sum of the windowed camera pixel stream (Unit III), in which ‘Din’ represents for data input of each line, and ‘sum’ the sum of the pixels in each line.

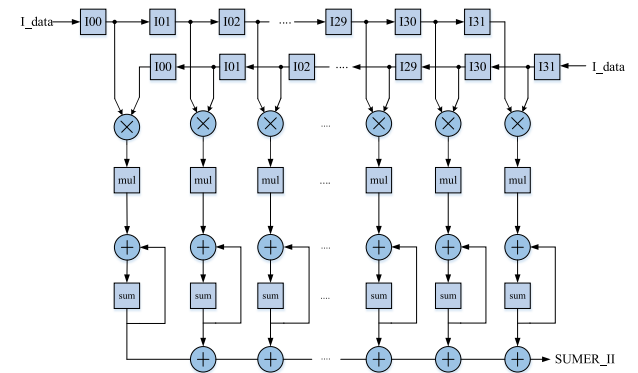


FIGURE 8. Circuit architecture for squared sum of the windowed camera pixel stream (Unit V).

image stream are multiplied each other in a concurrent mode, and sum-addition reuse is adopted for the final squared sum. This kind of circuit structure can largely reduce consumption of adder resources with the aid of semi-concurrent processing mode.

The circuit architecture for the unit I is depicted in Fig. 9, in which one-row buffering pixels of the camera image stream is multiplied by a corresponding row buffering template pixels in a concurrent mode, and sum-addition reuse is adopted for the final correlation sum. Similarly, this kind of semi-concurrent processing can largely reduce hardware resource consumption.

D. MATCHING EXPERIMENTS

In the evaluation experiment, the size of the reference image is 640×480 , and the sizes of the template images are 20×20 , 32×32 , and 60×60 respectively. The ZNCC algorithm

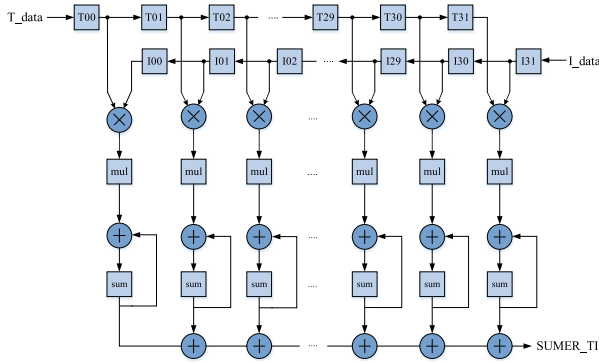


FIGURE 9. Circuit architecture for correlation computation between windowed camera stream image and template image.

runs on two kinds of evaluation platform, which are a personal computer (PC) platform (Intel i7-3770CPU@3.4GHZ, OpenCV image toolbox) and FPGA platform (Altera Cyclone V@5CSEMA5F31C6). Computation time in the FPGA is calculated by windowed image size, delay time and system clock frequency (50 MHz). The time consumption results are listed in Table 2. The comparisons of time consumption indicate that the presented ZNCC algorithm in FPGA run 15 ~ 36 times faster than that in PC, and keeping nearly unchanged under different template image size.

TABLE 2. Time consumption comparison (unit:ms).

Image size	Template size (pixel)	PC (SAD)	FPGA (SAD)	PC (ZNCC)	FPGA (ZNCC)
	20×20	70	5.7173	85	5.7178
640×480	32×32	82	5.4689	126	5.4696
	60×60	90	4.9118	276	4.9124

The hardware consumption comparisons on an Altera Cyclone V@5CSEMA5F31C6 between full current and concurrent with a multiplication-accumulate (MAC) circuits for ZNCC are listed in Table 3 under the same environment: reference image size 640 × 480 and template image size 32 × 32. The result indicates that the presented concurrent architecture for ZNCC consumes less FPGA hardware resources.

In order to evaluate our method with other competitive FPGA-based ZNCC method [25], we compare their hardware resource consumptions on a same Stratix IV FPGA (EP4SE530H35C2) used in [25]. The results are listed in Table 4. Obviously our method consumes fewer hardware resources than the ZNCC circuit in [25] under non-DSP and DSP modes. The reasons are that the correlation window computations are implemented by MAC circuit in our designed circuit but feedback FIFOs in [25]’s circuit, and windowed image is projected by register-based pipelining technology in our method but by memory-based pipelining technology.

TABLE 3. Resource consumption comparisons between concurrent and our presented ZNCC circuits.

Type	Total resources	Full concurrent		Our method	
		Used	Ratio (%)	Used	Ratio (%)
ALMs	32070	29173	91	6328	20
Block memory bits	4065280	303345	7.5	303345	7.5
PLLs	6	1	17	1	17
DSP blocks	87	69	79	69	79

TABLE 4. Resource consumption comparisons between the ZNCC circuits by our method and [25]’s method.

Type	Total resources	Our method	[25]’s method	Our method (DSP)	[25]’s method (DSP)
		Used (Ratio)	Used (Ratio)	Used (Ratio)	Used (Ratio)
DSP block	1024	23 (2.2%)	39 (3.8%)	93 (9%)	183 (17.9%)
Block memory bits	21233644	40960 (0.2%)	1295252 (6.1%)	40960 (0.2%)	1295252 (6.1%)
Logic registers	424960	8942 (2.1%)	11473 (2.7%)	10935 (2.6%)	8074 (1.9%)
ALUTs	424960	10328 (2.4%)	13173 (3.1%)	137175 (3.1%)	5524 (1.3%)

III. DESIGN ON FPGA-BASED ROBOTIC VISUAL TRACKING

A. DESIGN ON OPTICAL IMAGING SYSTEM

According to the requirements of compactness and large FOV of the optical imaging system, a novel compact back-view optical imaging system with a 45°-slant plane mirror is developed, which is used to eye-to-hand robotic visual tracking based on a Delta robot, as depicted in Fig. 10(a). Obviously, the working distance of the camera is S1 + S2, and the FOV size is W. In the proposed imaging system, the optical axis of the micro camera, which is connected with the FPGA board (DE1-SOC), is horizontal along with X-axis of the Delta robot. A 45°-slant plane mirror is mounted along the horizontal optical axis, as to make the optical axis turn to vertical orientation along Z-axis of the Delta robot. This kind of light path configuration enlarges the actual FOV under the limited installation space. Meanwhile, the captured image is background cluttered because the scattered light from the robot body can easily enter in the camera’ lens when looking at the object from down to up.

In Fig. 10(b), the moving target is hardly discriminated from the background. Facing this problem, we ingeniously add a front semi-transparent optical sheet in the light path, as shown in Fig. 10. From Fig. 10(c), the cluttered-background is successfully removed because of the semi-transparent sheet functions as a special nonlinear optical

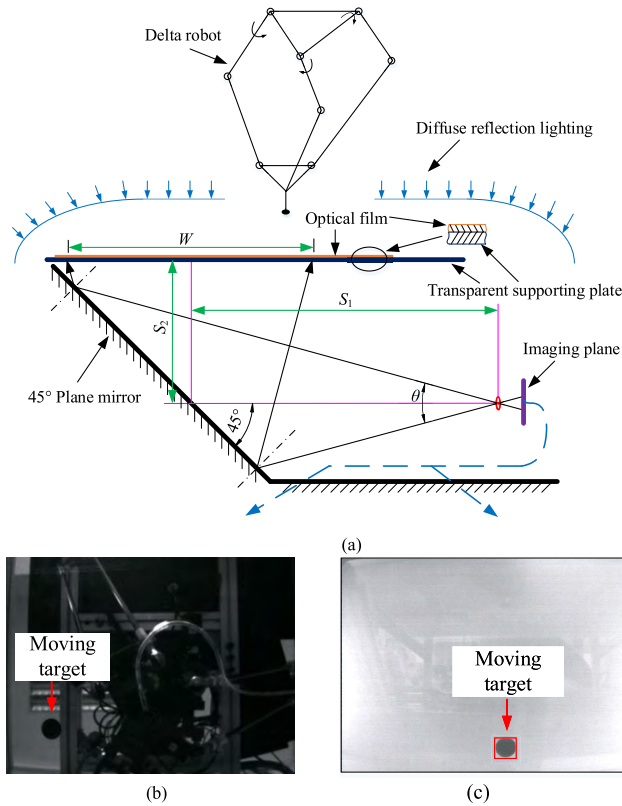


FIGURE 10. Imaging design of the robotic visual tracking system: (a) light path of the optical imaging; (b) the captured image without the semi-transparent optical film; (c) the captured image with the semi-transparent optical film.

filter. The filtered object image can largely reduce the burden of image processing on FPGAs.

B. ZNCC-BASED VISUAL TRACKING CORE ON FPGAs

The object position extraction computation is a key module in the robotic visual tracking system due to the requirements on real-time and high precision. We design a ZNCC-based object position extraction circuit on an Altera FPGA, shown in Fig. 11, which includes four main modules, including an image stream buffering core, a ZNCC-based template matching core, a VGA display core, and a UART data transmission core. The image stream buffering core functions as camera video stream buffering based on FIFO row buffering, and template image buffering based on shift register-based row buffering. The VGA display core can be used to output the captured image and the tracked red-boxed object image in real-time to a VGA monitor. The UART transmission core functions as data communication from the FPGA end to the PC end.

C. DELTA ROBOTIC VISUAL TRACKING CONTROL

Based on the presented ZNCC-based template matching core on FPGAs, a robotic visual tracking control with a Delta robot is proposed as shown in Fig. 12. The position error e between the position P_o of the tracked target and the end position P_t of the Delta robot is used to control the robot's

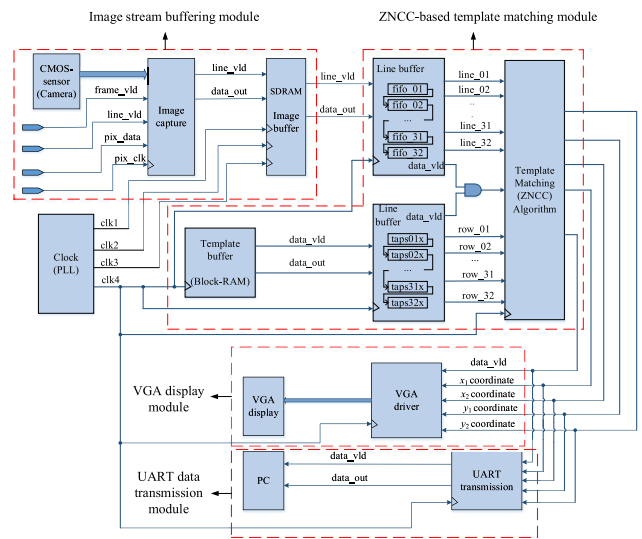


FIGURE 11. Grayscale template matching circuit on an FPGA.

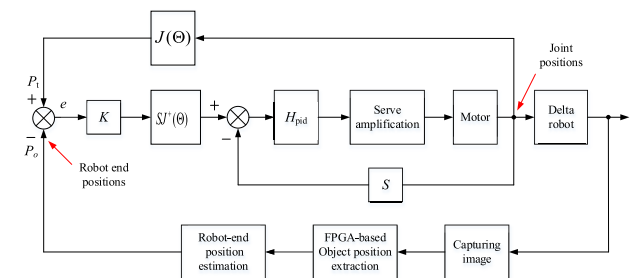


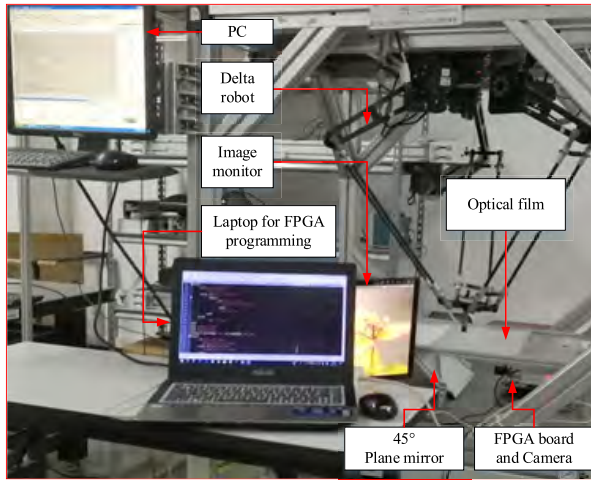
FIGURE 12. Block diagram of the robotic visual tracking control based on visual servoing, in which $J(\theta)$ and $J'(\theta)$ are the Jacobian and pseudo-inversed Jacobian.

motion by a proportional regulator. The positions P_o and P_t are calculated by the FPGA-based ZNCC core and the robot kinematics mapping respectively. The error e is mapped to the joint position error by the proportional gain K and inverse kinematics mapping. Meanwhile, the joint servo motors are controlled by a PID regulator H_{pid} in an interior loop. From Fig.12, the open transfer function can be simply treated as $K \times H_{pid}$ only considering the position loop and ignoring other loop units' influence due to the system's complexities. Because its control performance depends on the PID parameters, the system can be recognized in stability at a normal condition. But we will evaluate the control stability experimentally in Section IV.

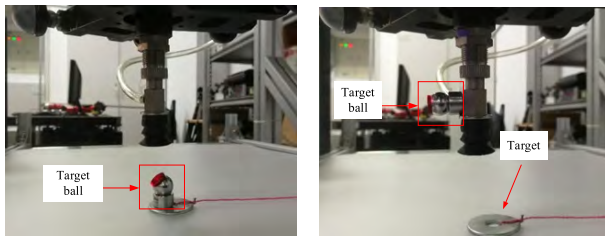
IV. EXPERIMENTS AND VERIFICATION

A. EXPERIMENTAL SETUP

In order to evaluate the presented robotic visual tracking methods, the different visual tracking experiments are implemented. The experimental setup is shown in Fig. 13(a), which is mainly composed of a Delta robot, an FPGA-based embedded vision imaging system, and an industrial control computer. The Delta robot was constructed by ourselves before, which controller is a PCI-based 4-axis motion controller in the industrial control computer. The FPGA-based embedded



(a)



(b)

(c)

FIGURE 13. The experimental setup of Delta robotic visual tracking system: (a) a whole configurations for object visual tracking; (b) object visual tracking experiment setup with a target reflection ball in a part view; (c) object visual tracking setup based on visual servoing in a partial view.

vision imaging system is constituted of a DE1-SOC FPGA development board with an integrated micro camera (752×480 MT9V034 with 4mm-focus lens), a 45° -slant plane mirror, a semi-transparent optical film, and a diffuse reflection light source. The object position extracted is transmitted to the PC through an RS232 interface.

Based on this experimental setup, a laser tracker AT901 by Leica is adopted for tracking performance evaluation in the visual tracking experiments. A target reflection ball is mounted on the object for the visual tracking experiment, shown in Fig. 13(b). A target reflection ball is mounted on the end of the robot for the visual tracking experiment based on visual servoing, shown in Fig. 13(c). In order to evaluate the dynamic tracking performance, a dynamical analyzer (DT9857E with QuickDAQ by DT) with a Kilter piezoelectric accelerator is adopted.

B. EXPERIMENTAL MATCHING COMPARISONS & DISCUSSION

Obviously, the uneven illumination phenomenon can be seen inevitably at the different location of the FOV of our presented imaging system from Fig. 10(c). Hence, we must evaluate the matching-based location performance of our chosen and designed ZNCC-on-FPGA method comparing with other FPGA-based methods. The ZNCC on a PC and SAD, NCC,

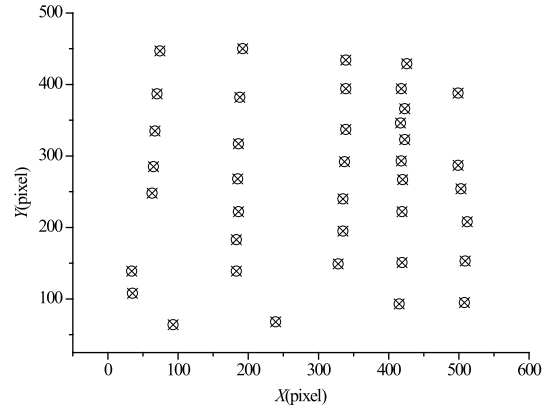


FIGURE 14. The 39 object matching points are set randomly on the FOV plane.

TABLE 5. Precision comparisons between the template matching on different platform.

Platform	PC (ZNCC)	FPGA (SAD)	FPGA (NCC)	FPGA (ZNCC)
Deviations	0	1.82	1.51	0.98

and ZNCC on FPGAs are used to implement object matching under an arbitrary object motion speed. We randomly designate the matching points as 39 object positions on the FOV plane, as shown in Fig. 14. The evaluating results are listed in Table 5, in which ZNCC on PCs is designated as a metric standard. From the comparisons, we can find that ZNCC-on-FPGAs method exhibit better object-locating performance than the other FPGA-based methods under uneven illumination.

In order to further evaluate the tracking performance, it is necessary to discuss the differences with another competing tracking method reported in [5]. The tracking algorithm in [5] is a kind of iteration-based optimization method, which searches the object in sub-regions of neighboring frames. Hence, the objection tracking in [5] is not strictly timing due to the unstable motion of the object and the non-real-time PC operation system, as may result in severe performance degrading. We evaluate the matching time with [5]’s results (Table 3 in [5]) under the same frame image 480×480 and template 100×100 sizes. Our running time is 3.65812 ms less than the average time cost 4.2 ms of AIOS in [5]. Obviously, our presented FPGA-based ZNCC method runs faster and more stable than AIOS in [5].

C. REAL-TIME OBJECT TRAJECTORY TRACKING

In order to evaluate the object tracking performance of ZNCC-based template matching on FPGAs independently, a visual tracking experiment without visual servoing is implemented. When an object is dragged and sliding on the glass plate with a semi-transparent film, the object position is computed and recorded in real-time by the FPGA-based ZNCC circuit and is measured by the laser tracker simultaneously. The visual tracking comparisons are shown in Fig.15. Obviously, our present ZNCC-based visual tracking error is less

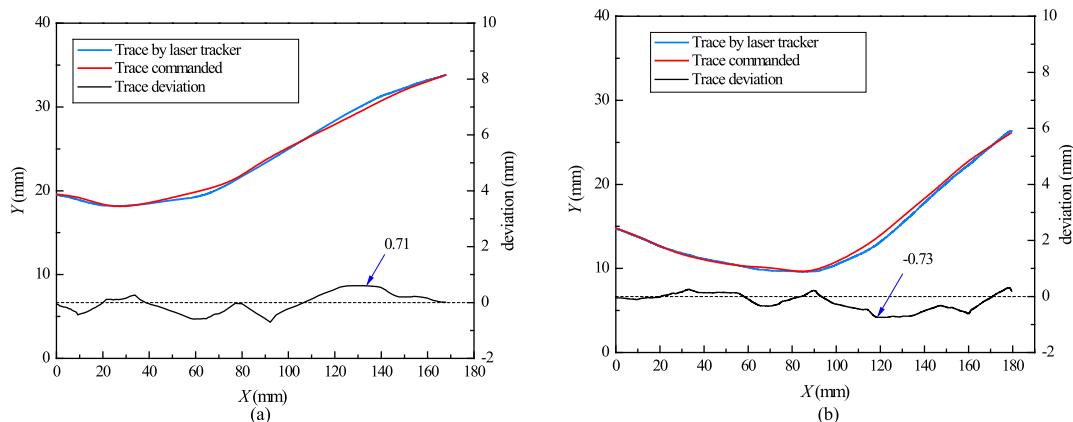


FIGURE 15. Comparisons of the recorded trajectories and their deviations when the maximum speed of the object motion speeds reach (a) 0.4m/s and (b) 0.7m/s.

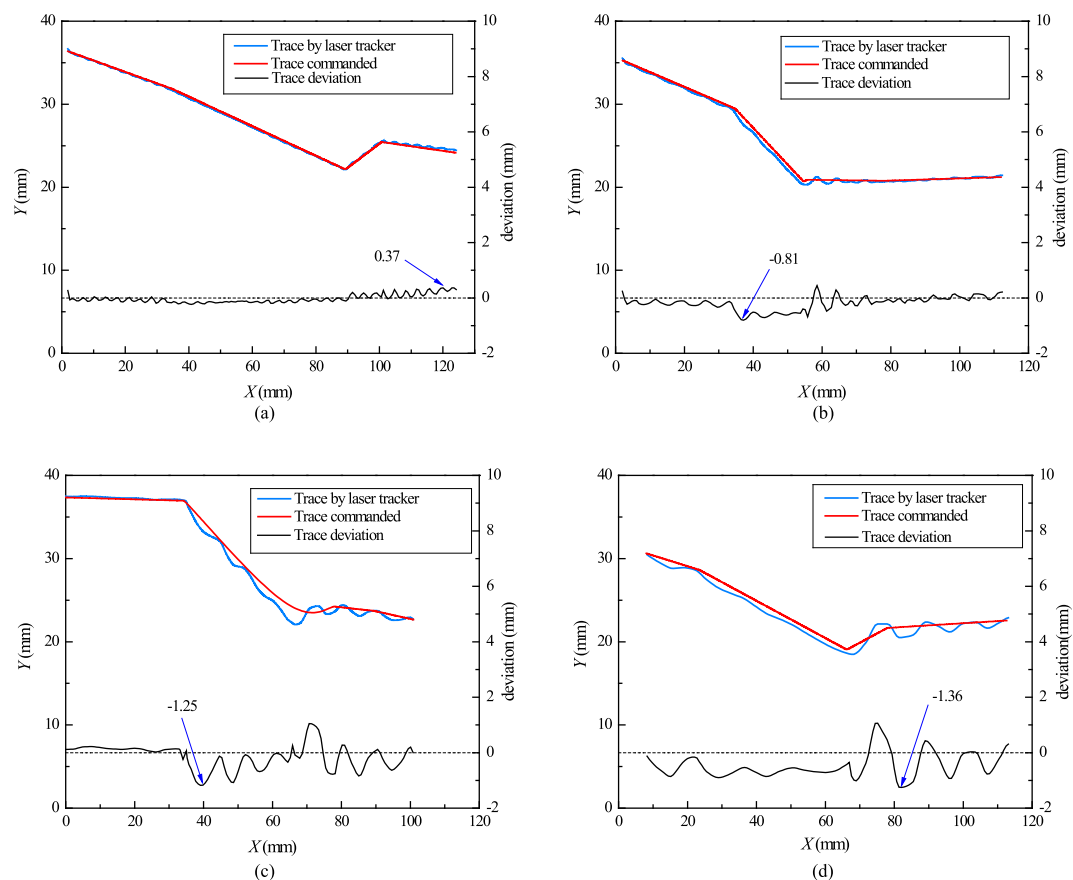


FIGURE 16. Robotic visual-servoing tracking performance: Actually measured tracking trajectories by our FPGA-based ZNCC method and the laser tracker, and their deviations when the object move in a speed of (a) 0.12m/s, (b) 0.26m/s, (c) 0.47m/s, and (d) 0.71m/s.

than 0.73 mm in object motion speed less than 0.7m/s when the motion trajectory is smooth.

D. REAL-TIME OBJECT VISUAL-SERVOING TRACKING

In order to evaluate the tracking performance based on visual servoing, the robotic visual-servoing tracking experiments under the different object motion speeds are implemented. The actual trajectories, recorded by our FPGA-based ZNCC

method and the laser tracker, and their deviations are depicted in Fig. 16. The four kinds of speed tracking trajectories all are arbitrarily commanded in the irregular traces, where turn sharply at some points. From the results, it is obvious that the tracking deviations increase with the object motion speed, and especially, the dynamical tracking deviation fluctuate largely when the object swerve sharply. But based on the visual servoing control, the tracking deviations tend to motion

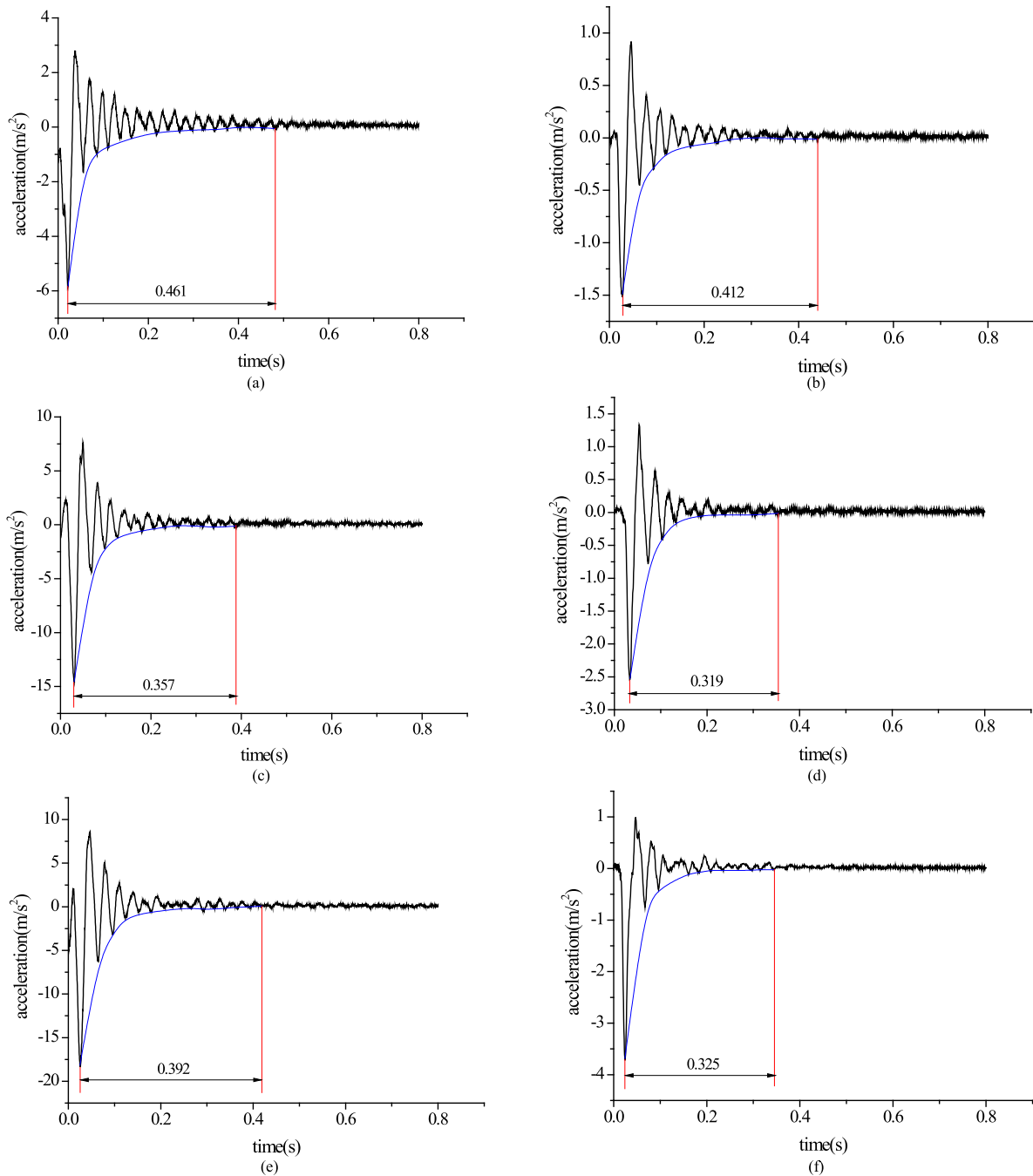


FIGURE 17. Dynamical performance of the robotic visual-servoing tracking: The measured vibration acceleration of the end of the Delta robot when stopping sharply in an open loop control mode at the initial motion speed (a) 0.26 m/s, (c) 0.47 m/s, and (e) 0.71 m/s; that in a visual-servoing close loop control mode at the initial motion speed (b) 0.26 m/s, (d) 0.47 m/s, and (f) 0.71 m/s.

steadily after the sharp-turning points. This verifies the presented FPGA-based ZNCC robotic visual tracking method is efficient.

E. DYNAMICAL PERFORMANCE OF REAL-TIME TRACKING

In order to further evaluate the dynamical visual-servoing tracking performance, we design a kind of special acceleration-based dynamical experiments. This dynamical experiment requires the end of the Delta robot is commanded

to move to a defined position along a linear trajectory at three kinds of motion speeds. When the robot stops, the vibration acceleration temporal signals are recorded simultaneously using a dynamical analyzer described in Subsection A. Meanwhile, three motion speed dynamical experiments are implemented under an open loop and visual-servoing close loop control modes respectively. The vibration acceleration temporal signals are drawn in Fig. 17. The reduced stabilization time is listed in Table 6 and 7. From the results, it is obvious

TABLE 6. Dynamical performance when stopping sharply in an open loop control mode.

No.	Speed (m/s)	Maximum acceleration (m/s ²)	Stabilization time (s)
1	0.26	-5.86	0.461
2	0.47	-14.42	0.357
3	0.71	-18.15	0.392

TABLE 7. Dynamical performance when stopping sharply in a visual-servoing close loop control mode.

No.	Speed (m/s)	Maximum acceleration (m/s ²)	Stabilization time (s)	Reduced time (s)
1	0.26	-1.49	0.412	0.049
2	0.47	-2.52	0.319	0.038
3	0.71	-3.86	0.325	0.067

that the maximum vibration accelerations and stabilization times in the closed loop control mode are all smaller than those in the open loop. This phenomenon demonstrates that visual servoing control successfully suppresses the impact-induced residual vibration, and improves the dynamical tracking performance. This can be explained using Nyquist sampling theorem, as the sampling frequency 60 Hz is larger than two times of the first order natural frequency of the Delta robot. The actual sampling frequency is limited by the GPIO interface to the micro camera. When using high-speed low-level camera interface, such as Cameralink, the dynamical performance can be further improved largely if measurement precision of the object tracked is increased simultaneously.

V. CONCLUSIONS

Visual-servoing can provide an efficient solution for a Delta robot close loop control. This paper proposes a ZNCC-based grayscale template matching concurrent circuit with cascading MAC computation on FPGAs, which can largely reduce valuable hardware consumption on a FPGA. Based on the presented FPGA-based ZNCC method, we develop a robotic visual-servoing tracking system using a Delta robot. In order to overcome target-matching issue due to background cluttered image, we design a compact optical imaging system with 45°-slant plane mirror and front optical filter film in the optical path. This largely lowers the complexity of the image preprocessing circuit in the FPGA. Various robotic visual trajectory tracking and dynamical visual-servoing tracking experiments are implemented. The experimental results verify the effectiveness in robotic visual tracking, and the presented visual-servoing tracking method has a better tracking precision and dynamical tracking performance than those in an open loop control mode.

REFERENCES

- [1] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, Nov. 2011.
- [2] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

- [3] F. Janabi-Sharifi, L. Deng, and W. Wilson, "Comparison of basic visual servoing methods," *IEEE/ASME Trans. Mechatronics*, vol. 16, no. 5, pp. 967–983, Oct. 2011.
- [4] F. Liu, C. Gong, X. Huang, T. Zhou, J. Yang, and D. Tao, "Robust visual tracking revisited: From correlation filter to template matching," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2777–2790, Jun. 2018.
- [5] H. Li, B. Zhu, Z. Chen, and X. Zhang, "Realtime in-plane displacements tracking of the precision positioning stage based on computer micro-vision," *Mech. Syst. Signal Process.*, vol. 124, pp. 111–123, Jun. 2019.
- [6] P. Cheng and C.-H. Menq, "Visual tracking of six-axis motion rendering ultraprecise visual servoing of microscopic objects," *IEEE-ASME Trans. Mechatronics*, vol. 23, no. 4, pp. 1564–1572, Aug. 2018.
- [7] S. Liu and Y. F. Li, "Precision 3D motion tracking for binocular microscopic vision system," *IEEE Trans. Ind. Electron.*, to be published. doi: 10.1109/TIE.2019.2893849.
- [8] Z. Chen, D. Zhou, H. Liao, and X. Zhang, "Precision alignment of optical fibers based on telecentric stereo microvision," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 4, pp. 1924–1934, Aug. 2016.
- [9] J.-H. Jean and F.-L. Lian, "Robust visual servo control of a mobile robot for object tracking using shape parameters," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 6, pp. 1461–1472, Nov. 2012.
- [10] Y.-W. Tu and M.-T. Ho, "Design and implementation of robust visual servoing control of an inverted pendulum with an FPGA-based image coprocessor," *Mechatronics*, vol. 21, no. 7, pp. 1170–1182, Oct. 2011.
- [11] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Trans. Image Process.*, vol. 9, no. 2, pp. 287–290, Feb. 2000.
- [12] X. Q. Gao, C. J. Duanmu, and C. R. Zou, "A multilevel successive elimination algorithm for block matching motion estimation," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 501–504, Mar. 2000.
- [13] W. Li and E. Salari, "Successive elimination algorithm for motion estimation," *IEEE Trans. Image Process.*, vol. 4, no. 1, pp. 105–107, Jan. 1995.
- [14] J. P. Lewis, "Fast template matching," *Vis. Interface*, vol. 95, pp. 15–19, May 1995.
- [15] T. Wu and A. Toet, "Speed-up template matching through integral image based weak classifiers," *J. Pattern Recognit. Res.*, vol. 1, pp. 1–12, Jan. 2014.
- [16] L. Di Stefano, S. Mattoccia, and F. Tombari, "ZNCC-based template matching using bounded partial correlation," *Pattern Recognit. Lett.*, vol. 26, no. 14, pp. 2129–2134, Oct. 2005.
- [17] O. Rahnama, D. Frost, O. Miksik, and P. H. Torr, "Real-time dense stereo matching with ELAS on FPGA-accelerated embedded devices," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2008–2015, Jul. 2018.
- [18] T. Adiono, M. D. Adhinata, N. Prihatiningrum, R. Disastra, R. V. W. Putra, and A. H. Salman, "An architecture design of SAD based template matching for fast queue counter in FPGA," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Phuket, Thailand, Oct. 2016, pp. 1–4.
- [19] R. A. Hamzah and H. Ibrahim, "Literature survey on stereo vision disparity map algorithms," *J. Sensors*, vol. 2016, Nov. 2015, Art. no. 8742920.
- [20] W. Wang, J. Yan, N. Xu, Y. Wang, and F.-H. Hsu, "Real-time high-quality stereo vision system in FPGA," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 10, pp. 1696–1708, Oct. 2015.
- [21] P. Elfert, T. Tiemerding, C. Diederichs, and S. Fatikow, "Advanced methods for high-speed template matching targeting FPGAs," in *Proc. Int. Symp. Optomechatronic Technol.*, Seattle, WA, USA, Nov. 2014, pp. 33–37.
- [22] A. Alabdo, J. Pérez, G. J. García, J. Pomares, and F. Torres, "FPGA-based architecture for direct visual control robotic systems," *Mechatronics*, vol. 39, pp. 204–216, Nov. 2016.
- [23] J. Pérez, A. Alabdo, J. Pomares, G. J. García, and F. Torres, "FPGA-based visual control system using dynamic perceptibility," *Robot. Comput. Integr. Manuf.*, vol. 41, pp. 13–22, Oct. 2016.
- [24] C. Diederichs, "Fast and robust camera-based motion tracking using FPGAs in microrobotics," Ph.D. dissertation, Dept. Comput. Sci., Univ. Oldenburg, Oldenburg, Germany, 2014.
- [25] E. S. Albuquerque, A. P. A. Ferreira, J. G. M. Silva, J. P. F. Barbosa, R. L. M. Carlos, D. S. Albuquerque, and E. N. S. Barros, "An FPGA-based accelerator for multiple real-time template matching," in *Proc. 29th Symp. Integr. Circuits Syst. Design*, Belo Horizonte, Brazil, Aug./Sep. 2016, pp. 1–6.
- [26] A. Lindoso and L. Entrena, "High performance FPGA-based image correlation," *J. Real-Time Image Process.*, vol. 2, pp. 223–233, Dec. 2007.

- [27] Q. Zhou, L. Yang, and H. Cao, "A configurable circuit for cross-correlation in real-time image matching," *J. Comput. Sci. Technol.*, vol. 32, pp. 1305–1318, Nov. 2017.



ZHONG CHEN was born in Chengdu, Sichuan, China, in 1968. He received the B.Eng. degree from the Chengdu University of Science and Technology, in 1990, and the M.Sc. and Ph.D. degrees from the South China University of Technology, in 1996 and 2003, respectively.

From 1993 to 1996, he was with Sichuan Chemical Factory. From 2015 to 2016, he conducted some collaborated research as a Visiting Scholar at the Division of Microrobotics and Control Engineering, Oldenburg University, Germany. Since 2003, he has been the Deputy Director of the Guangdong Provincial Key Laboratory of Precision Equipment and Manufacturing Technology and has been taking parts in a research team led by Prof. X. Zhang. Since 2005, he has been an Associate Professor and a Professor with the School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou, China. His research interests include precision measurement and testing, dynamics of compliant mechanisms, and machine vision and its application.



SHUAI LI was born in Xiangyang, Hubei, China, in 1991. He received the B.Eng. degree from Wuhan Textile University, in 2016. He is currently pursuing the master's degree with the Guangdong Provincial Key Laboratory of Precision Equipment and Manufacturing Technology, School of Mechanical and Automotive Engineering, South China University of Technology. His research interests include field-programmable gate array (FPGA)-based embedded machine vision, digital image processing, and communication technology.



NING ZHANG was born in Chongqing, China, in 1998. He is currently pursuing the bachelor's degree in mechatronics with the School of Mechanical and Automotive Engineering, South China University of Technology. His research interests include pattern recognition and control theory.



YUYA HAO was born in Yangquan, Shanxi, China, in 1995. He received the B.Eng. degree from the Guangdong University of Technology, in 2018. He is currently pursuing the master's degree with the Guangdong Provincial Key Laboratory of Precision Equipment and Manufacturing Technology, School of Mechanical and Automotive Engineering, South China University of Technology. His research interests include machine vision and precision measurement.



XIANMIN ZHANG was born in Zhuozhou, Hebei, China, in 1964. He received the Ph.D. degree from the Beijing University of Aeronautics and Astronautics, in 1993.

Since 2003, he has been a Professor with the South China University of Technology. He was the Dean of the School of Mechanical and Automotive Engineering, South China University of Technology, in 2013. His research interests include precision positioning, mechanism design and dynamics, and inspection technique.

...