# Depth Estimation Based on Pyramid Normalized Cross-correlation Algorithm for Vergence Control

Mohamed, A

http://hdl.handle.net/10026.1/12964

10.1109/ACCESS.2018.2877721
IEEE Access
Institute of Electrical and Electronics Engineers (IEEE)

# Pyramid Normalized Cross-correlation-based Algorithm for Vergence Control

# Develop an Active Binocular Platform Controller based Pyramid Normalized Cross-Correlation for Harvesting Process

**Abdulla Mohamed[*1], Phil F. Culverhouse[1], Angelo Cangelosi[2], and Chenguang Yang[3]**
[1] Centre for Robotics & Neural Systems, Plymouth University, UK
[2] School of Computer Science, University of Manchester, UK
[3] Zienkiewicz Centre for Computational Engineering, Swansea University, UK

Corresponding author: Abdulla Mohamed (e-mail: abdulla.mohammad@plymouth.ac.uk).

**ABSTRACT** A depth estimation algorithm based on vergence vision using a mechanical joint attached to two cameras is proposed. A Gaussian pyramid template-matching approach is used to align the view of the slave camera to the fixation point of the master camera. The master camera uses an object detection algorithm to find the target's centroid and centers it relative to the image coordinates. Then, vergence movement of the slave camera is performed using a pyramid normalized cross-correlation algorithm. Simple geometric triangulation is employed to compute the depth of that target. This proposed method was implemented using an active binocular vision platform with five degrees of freedom where four degrees of freedom to control the pan and tilt independently, and one degree of freedom to control the baseline which is the distance between the camera. This system was designed for implementation in agriculture harvesting applications. Analysis of field trial results indicates a worst-case precision of a target tomatoes' depth to be $\pm$1.32 cm at a depth of 85 cm.

**INDEX TERMS** Active stereo vision, Image pyramid, Template-matching, Vergence vision

## I. INTRODUCTION

Depth perception refers to the ability to perceive the world in three dimensions. Among animals, different visual systems use different depth perception mechanisms. In this context, motion parallax involves moving the eye or head from side to side and adjusting the focal length to obtain a sharp focus [1]. Some animals use correspondence wherein a feature perceived by the left eye is matched to the same feature perceived by the right eye to determine differences [2], and other animals, including humans, use vergence cues to measure depth [3]. However, in humans, the assignment of the master eye appears to depend on the scene and target object [4]. Like humans, many animals use different vision cues simultaneously [5].

Stereoscopy techniques have been widely used to measure the distance by focusing two devices, typically cameras, at different positions on a single point. Here, the distance between the devices is known; thus, a triangle is formed among both camera and the point. Geometric triangulation is used to calculate the distance between a camera and the target point. Disparity calculation between two fixed cameras is an alternative; however, this method suffers known associated errors [6], [7], [8]. A previous study [9] employed a block matching algorithm to investigate the error generated by the disparity calculation. This study explores errors associated with vergence cue in-depth estimation for a pair of cameras with a variable baseline [10].

Vergence vision algorithms to control the verge angle between a master and a slave camera can be categorized as (1) verge-based feature matching using correspondence, (2) verge-based local phase differences and (3) verge-based local correlation [11]. In the first category, an algorithm extracts features from disparity information to compute disparity and move the camera based on the computed disparity [12], [13]. Here, a representative example is the zero disparity filter (ZDF) [14]. The second category involves local phase differences. Here, images are converted to the Fourier domain to control the verge angle [11], [15]. The third category computes the correlation between left and right images using the sum of absolute differences (SAD) or normalized cross-correlation (NCC). Note that SAD is affected by light change; thus, the NCC algorithm is preferable [16], [17]. Some studies have controlled vergence by transforming the image to log-polar space where the image has a high resolution in the center and low resolution on the edges, which mimics how human eyes function [18], [19], [20].

Rougeaux et al. [21] built an active stereo vision system that used two joints to control two cameras to track moving objects against a complex background and developed the idea of a virtual horopter. Here, the horopter is the circle that passes through the center of both cameras' three-dimensional (3D) views and the target, wherein the target has zero disparity [22]. A virtual horopter was designed to continue tracking the object if it goes outside the actual horopter by shifting the image to the left or right to increase or decrease the size of the horopter. Here, the algorithm used is ZDF-based edge detection. The system used by Rougeaux et al. operated at an image resolution of 230 × 130 × 8 bits and 30 FPS. The accuracy of controlling the pan angle was ±0.4°, and the system could track an object in the horopter circle at 50° per second. The depth error calculated at a distance of 112 mm was ±6 cm. Their experiment demonstrated that this shifting approach enabled a virtual horopter to track objects. However, shifting the image can lead to error because the baseline is fixed and the image is taken at the baseline distance. When the image is shifted, the baseline increases, which was not considered in the final depth calculation.

Dankers et al used active stereo vision to track a hand using a maximum a posterior probability ZDF (MAP ZDF) [16]. Their platform used the ZDF to track and segment the hand gesture. The difference value between Gaussians and NCC were utilized in the algorithm to smooth images and compute the disparity map. Note that Dankers et al. used the same methods as Rougeaux et al. [21]. The MAP ZDF was used to track the hand, control the camera joint to follow the hand and maintain the center of gravity within the cameras' centers. This system operated at an average of 25 FPS and achieved ample working space. The object tracking performance of this system was robust and accurate; however, the system occasionally failed to track fast moving hands. Dankers et al. employed the same standard methods as Rougeaux et al. [21], i.e., both studies used the ZDF at the same resolution and the speed was nearly the same (25 and 30 FPS, respectively). However, the method proposed by Rougeaux et al. was faster and more reliable than that in the software process Dankers et al. used to segment the hand.

Marefat et al. [15] evaluated gaze stabilization for object tracking to calculate the disparity map and improve the fixation point of the slave camera. Their system employed two cameras with independent pan and tilt control. The focus of this work was to control the verge angle. Marefat et al. investigated the disparity of the object of interest rather than exploring the disparity of the entire scene. To calculate disparity, a Fourier phase-based approach was used to distinguish the subject matter between both images. This transformation was estimated by calculating the phase difference between the images. The transformation process was performed in binary space to increase computation speed. Their results indicate that the disparity error depends on the vergence angle, which is similar to the previously mentioned studies on the fixation point. Gibaldi et al. [11] employed local phase differences between the left and right images to control the verge angle. Here, a Fourier-shift theorem based on a population of oriented disparity detectors was applied in a two-dimensional (2D) search, where disparity was viewed from both horizontal and vertical views.

The search speed of matching feature-based correlation vergence control has been improved by implementing a coarse-to-fine pyramid algorithm [17], [23]. Zhang and Tay [17] reported a solution based on a coarse-to-fine (pyramid images) search algorithm where the image is transformed to log-polar space prior to constructing the pyramid. An NCC search algorithm was run on the log-polar images, and the output was used to set the vergence angle. Implementing this log-polar transformation into the search algorithm improved tracking performance in a compound sense with multiple disparities. In Zhang and Tay's [17] study the image resolution was 200 × 200, the window size was 84 × 84 and the baseline was 24 cm. Here, a three-level pyramid was used. The average accuracy of the depth measurement for different objects was 90%.

Different from such previous methods, this study proposes a depth estimation method that uses a vergence controller. In the proposed method, a matching feature-based correlation technique is integrated with a Gaussian pyramid to control the slave camera's gaze and integrated with exciting platform. Here, depth is estimated considering target position by analyzing the external parameters of the system such as the baseline and the pan angle of the left and right camera. The point where both focal rays intersect is referred to as the fixation point. An algorithm was developed to ensure that both rays intersect at the fixation point. In addition, a fast control system was developed to keep both cameras focused on the same point. We performed experiments to evaluate the proposed method, and the results are compared to those reported by Zhang and Tay [17].

Moreover, the system has tested and evaluated in the field. We believe that a binocular based vergence controller has never been used in tomato fruits harvesting application.

The remainder of this paper is organized as follows. Verge depth mathematics is introduced in Section 2. Section 3 describes the experimental setup used to evaluate the performance of the proposed algorithm and the implemented platform. Experimental results are presented and discussed in Section 4, and conclusions and suggestions for future work are given in Section 5.

## II. BACKGROUND: VERGE-BASED DEPTH VISION

Vergence cue methods focus on a point in space where the centers of two images ($I_{c_l}$ and $l_{c_r}$) are aligned with the interested point of the target, i.e., the gaze point. The angle between the two cameras is the vergence angle. Figure 1 shows the coordinate frames of the system. The fixed frame, i.e., the center of the system, is denoted $O_P$, and the origin of the frames is denoted $O_i$, where $i = r\,(i = l)$ is the right (left) frame. The origins of the left and right frames are coincident on the $Y_p$-axis, and the distance between the origin of left and the origin of right and parallel to the $Y_p$ is the baseline $B$, which is changeable. Note that the right and left frames rotate around $Y_r$ and $Y_l$, respectively.

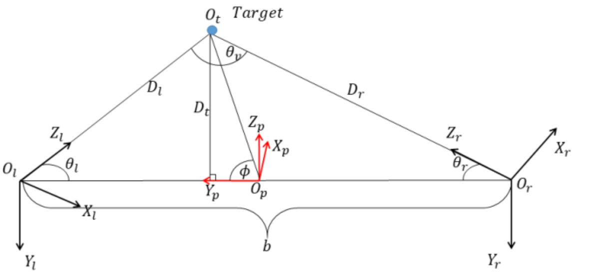


**Figure 1.** **Two-and-a-half depth coordinate system.**

The geometric rotation between the left and right frames is referred to as an essential matrix $E_{lr}$, which is a $(4 \times 4)$ homogenous transformation matrix [24]. $E_{lr}$ describes the rotation and translation between the left and right frames expressed as follows:

$$E_{lr} = [R\,|T] \tag{1}$$

Here, $R$ is the Euler rotation angle of the right frame relative to the left frame, and $T = [\,X\;Y\;Z]^T$ is the translation of the right frame relative to the left frame. Each frame rotates independently; thus, $E_{lr}$ is the product of $E_l$ and $E_r$ calculated from $O_p$. Note that the left and right frames only rotate around the $Y_i$-axis; therefore, the only variable rotation matrix used is $R(Y)$, where $\theta_i$ is the angle of rotation. $R(Y)$ is expressed as follows:

$$R(Y)_i = \begin{bmatrix} \cos(\theta_i) & 0 & \sin(\theta_i) \\ 0 & 1 & 0 \\ -\sin(\theta_i) & 0 & \cos(\theta_i) \end{bmatrix} \tag{2}$$

The essential matrix $E_{lr}$ contains the information required to calculate the distance to the target, i.e., the angles of rotation and the distance between the frames [22]. Assuming that the plane $(O_r, O_l, O_t)$ intersects the target origin, the left origin and right origin, all internal angles (including that of the baseline) can be derived trigonometrically. Therefore, the internal angles can be expressed as follows:

$$\theta_l = 90 - \theta_{external_l} \tag{3}$$

$$\theta_r = 90 + \theta_{external_r} \tag{4}$$

$$\theta_{verge} = 180 - (\theta_l + \theta_r) \tag{5}$$

The sine rule is used to calculate $D_l$ and $D_r$ as follows:

$$D_l = \frac{b \times \sin(\theta_r)}{\sin(\theta_{verge})} \tag{6}$$

$$D_r = \frac{b \times \sin(\theta_l)}{\sin(\theta_{verge})} \tag{7}$$

Therefore, to calculate the depth of the target on plane $(O_p\, O_l\, O_r)$, a line $D_t$ is drawn from the target $O_t$ perpendicular to $X_p$ (Figure 1) to form two right angles (left: $(O_l\, O_t\, O_p)$; right: $(O_r\, O_t\, O_p)$). Then use Pythagorean theorem, $D_t$ is calculated using Eq. (8) where $i = r\,, l$.

$$D_t = D_i \times \sin(\theta_i) \tag{8}$$

Now use Pythagorean theorem to calculate $Y_t$ as well, as shown in Eq. (9). Here, baseline $b$ is negative when $\theta_l$ is used in the calculation and positive when $\theta_r$ is used.

$$Y_t = \frac{\pm b}{2} \pm \left(D_i \times \cos(\theta_i)\right) \tag{9}$$

Eqs. (8) and (9) give the position of the target relative to the platform's frame. Note that the sign of the baseline depends on which side is considered to compute $Y_t$ (e.g., for the left angle, $Y_t = \frac{b}{2} - \left(D_i \times \cos(\theta_i)\right)$).

Note that the above discussion assumes that the target lies on the same plane as the image center. Therefore, using the rotations angle of a tilting motor, the final coordinates of the object can be calculated. In standard coordinates, $D_t$ is the $X$-axis and $Y_t$ is the $Y$-axis; therefore, the $Z$-axis is computed using the rotation around the tilting axis $R_y(\theta)$ (Eq. (10)).

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) \\ 0 & 1 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix} \begin{bmatrix} D \\ Y_t \\ 0 \end{bmatrix} \tag{10}$$

### A. *FIXATION OBJECT*

The platform includes a master and slave system to keep both cameras' centers focused on the target centroid using coarse-to-fine template matching based on a Gaussian pyramid where the slave tracks the center of the master camera.

**Figure 2.** **ArUco pattern refers to number 1.**

Typically, the master camera tracks the target using an object detection algorithm or color threshold technique. In this study, the ArUco detection algorithm, which tracks a specific target (Figure 2), is used. ArUco is an OpenCV library for camera pose estimation using squared markers [25]. The ArUco algorithm was selected to evaluate the system because it demonstrates high precision and fast pattern detection.

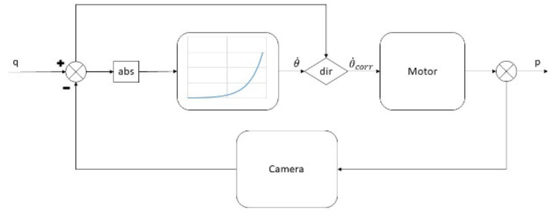


**Figure 3.** **Motor controller based on the exponential function.**

The vergence controller uses an exponential function controller to align the centroid of the target to the center of the image (Figure 3). The PNCC algorithm return the position of the target in image coordinate where the center of the image is where the fixation point is required to be on. This difference is describing how far the object from the center of the image. This difference is the input to the exponential function and the output is angular velocity. The input and the output of the exponential function is always positive therefore, a direction correction function is used to correct the direction of the angular velocity.

$$\dot{\theta} = \exp^{(q \times \lambda)} \times \beta \tag{11}$$

Where $\dot{\theta}$ is the angular velocity in $rpm$, $\beta$ is control constant that control the range of the output to meet the range of the motor. While $\lambda$ is the constant describe the shape of the output. $q$ is the input to the exponential function where this has to be always positive. The direction of the output velocity $\dot{\theta}$ is corrected using the sign of the $q$ before taking the absolute of this value eq.(12).

$$\dot{\theta}_{corr} = \frac{q}{|q|} \times \dot{\theta} \tag{12}$$

Where $|q|$ is an absolute q always positive.

## B. COARSE-TO-FINE TEMPLATE-MATCHING ALGORITHM

A template-matching algorithm searches a large image $I$ using a small image template $T$, where the template represents the target information in the image. Here, the search type is classified as 2D, i.e., the template passes over an image in two directions $u$ and $v$ that represent the *x*- and *y*-axes, respectively. The similarity between the template and the region in the image is then computed. Therefore, to determine the position of the template in image $I$, a cost function is used to estimate similarity between the template and the position of the template in the image, which is stored in matrix $M$. The size of matrix $M$ is determined by subtracting the size of template $T$ from image $I$ $(I - T)$. Depending on the cost function, the location of the template in the image is determined by the smallest or largest value in $M$. Eq. (13) shows an NCC differences cost function where the large image is $(u, v)$ and the size of the template is $(m, n)$. Using this cost function, the best location for the template is determined by the largest value in $M$.

$$NCC(u, v) = \frac{\sum_{m,n}(T(m,n) \times I(u+m, v+n))}{\sqrt{\sum_{m,n} T(m,n)^2 \times \sum_{m,n} I(u+m, v+n)^2}} \tag{13}$$

NCC methods were selected to compute matching because NCC removes large differences between the template and image around a large brightness [26].

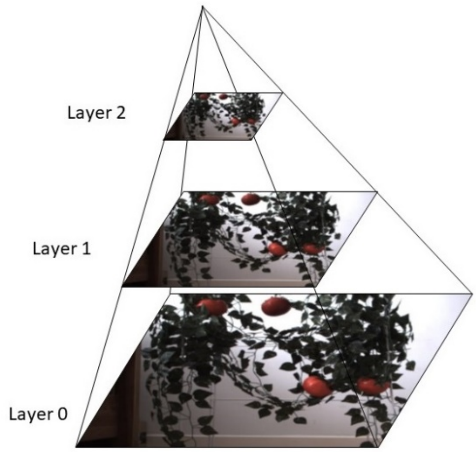


**Figure 4.** **A three-level Gaussian pyramid.**

In the proposed platform, the template is a window with a small number of pixels cropped from the center of the master image (100x100). A coarse-to-fine Gaussian pyramid is constructed to improve tracking quality and increase the speed of the template-matching algorithm by reducing the number of unnecessary computations (Figure 4). The Gaussian pyramid algorithm produces a sequence of images by down-sampling, where the image size and resolution are reduced at every level ($I^{Lth}$) [27]. Here, the Gaussian pyramid produces images that are half the size of the level under the current level ($I^{L+i} = \frac{I^L}{2}$).

Note that Gaussian pyramids are calculated for both the template and image. The cross-correlation algorithm implemented at the top level $I^L$ (coarse) uses both the template and image. A threshold is applied to the output $M^L$. $M^L$ is up-sampled by a factor of 2, and a counter is applied to find the region with large differences $C^L = (x,\ y,\ width,\ height)$. The search in level $I^{L-1}$ is constrained to region $C^L$, which covers the neighborhood pixels around the maximum likeliness found in the previous level. In each level, the search is reduced to the highest matching feature until the search reaches the base of the pyramid $I^0$,which represents the finest resolution image.

## C. SENSITIVITY OF DEPTH MEASUREMENT TO ERRONEOUS SYSTEM ASSEMBLY AND CALIBRATION

In practical applications, various parameters must be considered to generate accurate output. These parameters can have significant effect on the system's output depending on their contribution to the overall error. Kanatani [6] identified four factors that influence the system's final output, i.e., (1) accuracy bound, (2) reliability evaluation, (3) computational efficiency and (4) model plausibility.

In a binocular vision system, different parameters contribute to depth measurement errors. A detailed discussion of these parameters can be found in the literature [6], [7]. Such parameters vary relatively to the extent to which they increase the number of errors in the depth measurement. In practical vergence vision applications, the verge angle depends on the fixation point and the gaze of slave camera having the same fixation point as the master camera. Computing the depth depends on the verge angle, which, in turn, depends on encoder reading, baseline measurement and pixel size. Note that these values relate to the geometry of the platform and its manufacturing process. The accuracy of computing the fixation point depends on the computational efficiency and the algorithm used to compute the fixation point.

In the following, we provide a geometrical analysis of the system to evaluate parameters contribution to error.

By solving Eq. (8) using the left angle is as follows:

$$D_t = \frac{b\sin(2\theta_l)}{2\sin(\theta_v)} \tag{14}$$

By applying differentiation to compute $\delta D_t$ relative to baseline $b$, we obtain the following.

$$\delta D_t = \frac{b^2\sin(2\theta_l)}{2\sin(\theta_v)}\delta b \tag{15}$$

Now solving Eq. (14) to $b$, then replacing the output with $b$ in Eq.(15), we derive

$$\delta D_t = \frac{D_t^2\sin(2\theta_l)^3}{8\sin(\theta_v)^3}\,\delta b \tag{16}$$

Eq. (16) shows that depth error is proportional to the square of the depth. However, compared to orthogonal stereo vision, the depth error in a verge system is multiplied by eight. The same applies to Eq. (9), which shows that the size of the baseline does not affect error in the $Y_t$ measurement. Here, the verge angle is inversely proportional to the error in measuring $Y_t$.

To clarify the effect when a vergence angle error occurs in the system, we differentiate Eq. (14) relative to verge angle $\theta_v$. The output is shown in Eq. (17).

$$\delta D_t = \frac{b}{2}\sin(2\theta_l)\,(-\csc(\theta_v))\cot(\theta_v)\,\delta\theta_v \tag{17}$$

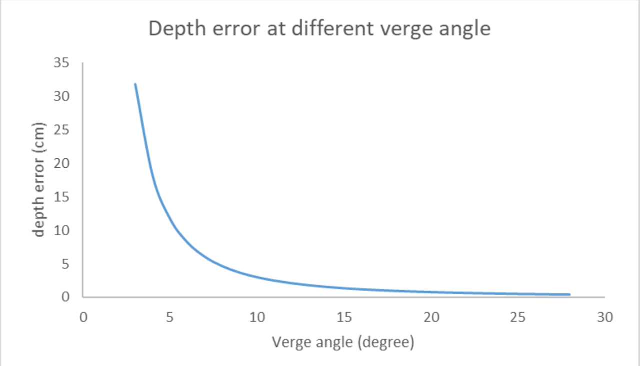


**Figure 5. Depth error at different verge angles (smaller angles give greater depth).**

Increasing the verge angle reduces the depth error $\delta D_t$ where the verge angle is inversely proportional to the square depth error. The verge angle is the sum of $\theta_r$ and $\theta_l$; therefore, both angles contribute to the accuracy of the final depth output. Figure 5 shows the theoretical error generated by the verge angle in a worst-case scenario for the proposed system where the encoder resolution is 12 bits (equal to 0.08°).

The depth error $\delta D_t$ shown in Figure 5 was computed using the maximum error generated using the given encoders. The profile of the error curve exactly follows the curve generated by the analytical error (Eq. (17)) [28]. The results of both verge angle and depth relative to depth error demonstrate that the size of the error increases significantly as the measured depth increases, which is consistent with the result obtained when a disparity calculation is used in an orthogonal stereo vision.

Disparity maps employ a multi-stage technique that introduces errors into the process, including the pixel matching process. An orthogonal stereo vision system introduces errors from different parameters that significantly affect the final output, such as misalignment of the $y$-axis of both cameras, camera rotation and the size of the pixels. Based on a 3D reconstruction of the target, Kanatani (2005) discussed errors in stereo vision systems and how they affect performance. An orthogonal stereo vision system depends on the quality of the calibration output, where internal and external parameters are computed for use in a rectification process. Error analysis of standard stereo vision systems can be classified as the effect on normalized image coordinates, the effect on pixel coordinates and the effect on 3D reconstruction [7]. In vergence stereo systems, depth errors are generated by the external geometry (i.e., the assembly and encoder quality) and the effectiveness of the tracking algorithm.

The source of errors in vergence and orthogonal stereo vision systems are compared in Table 1. As can be seen, most of the parameters of orthogonal systems do not contribute to errors in vergence systems.

TABLE I ERROR SOURCES IN ORTHOGONAL STEREO VISION VERGENCE VISION SYSTEMS

| Error Source | Orthogonal Stereo Vision | Vergence Vision |
|---|---|---|
| Baseline | $\delta D \propto \frac{D^2}{f * b} \delta d$ | $\delta D_t \propto \frac{D_t^2 \times \sin(2 \times \theta_l)^3}{8 \times \sin(\theta_v)^3} \delta b$ |
| Verge angle | $\delta D \; \alpha \frac{D^2}{b} \tilde{x}_L \, \tilde{y}_L \delta\theta_v$ | $\delta D_t \; \alpha - \csc(\theta_v)\cot(\theta_v)\,\delta\theta_v$ |
| Tilting angle | $\frac{\delta Z}{\delta \Theta_l} \approx \frac{Z^2}{b} \tilde{x}_l \tilde{y}_l$ | None |
| Focal length | $\frac{\delta Z}{\delta f} \approx \frac{Z}{f}$ | None |

Pixel size also contributes to the performance of the vergence controller, where smaller pixel size results in more accurate target centroid detection. In fixed-camera stereo vision systems, pixel size contributes to the size of the depth error when computing disparity. While active stereo vision depends on the precise of location of the centroid of the target.

### *D. MOTOR CONTROLLER-BASED EXPONENTIAL FUNCTION*

The control system used to move the pan and tilt motors are based on an exponential function. The control system comprises a motor and a camera that is used as a feedback sensor.

$$\dot{\theta} = \exp^{(q \times \lambda)} \times \beta \tag{18}$$

A block diagram of the motor controller is shown in Figure 3. Here, the camera generates the position of the target in pixel. The input to the motor is the angular velocity $\dot{\theta}$ in rpm, which is computed using the exponential function (Eq. (18)).

## III. EXPERIMENTS

Quantitative and qualitative experiments using a five degree of freedom active stereo vision platform [10] (Figure 6) were conducted to evaluate the proposed algorithm.

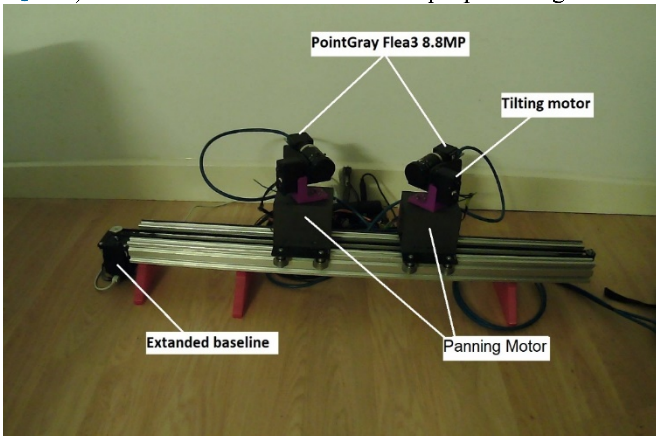


**Figure 6.** A stereo vision platform.

### *A. PYRAMID NCC EXPERIMENT*

**Figure 7.** A template for in-depth measurement with 16 targets at different heights.

The first experiment involved a qualitative test where a 40 × 40 cm template comprising 16 targets at different heights and positions with a center-to-center distance of 10 ± 0.1 cm was constructed (Figure 7). The targets were numbered from 1 to 16 according to the ArUco numbering system. The numbering started at the top left corner and ended at the bottom right corner. The origin of the template was marked as number 18. It is critical to allocate the center of the template because the targets' coordinates can only be obtained if the origin is known. As shown in Figure 7, the 16 targets are somewhat similar in terms of their intensity, and this similarity confuses the algorithm, which justifies using the vergence controller to track each target individually to determine how accurately the system verges toward the targets.

### *B. UNBALANCED BRIGHTNESS CONDITIONS*

**Figure 8.** A tomato setup used in evaluating the performance of the platform.

The second experiment was a qualitative and quantitative test that used an artificial setup for tomato fruit detection to evaluate the pyramid NCC (PNCC) algorithm and the platform (Figure 8) to test the performance of the system under different lighting inputs between the master and slave cameras and depth measurements. This experiment involved three different configurations, where the slave camera's IRIS status was fully open in all configurations. Here, the difference was only in the state of the master camera's IRIS, i.e., fully opened, one-quarter opened and half opened. There were four targets in these configurations with different depths and positions, where the master camera was manually fixed on the target. Then, the vergence algorithm was executed for the slave camera to verge on the target, whose depth was recorded. The position of the tomato was set within the calibrated range of the platform (minimum to maximum range: 40–200 cm). Note that the position of the tomato was determined using a measuring tape.

### C. DEPTH ESTIMATION EXPERIMENT

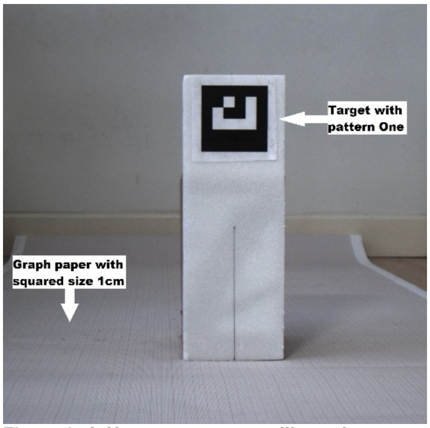


**Figure 9. ArUco pattern on a calibrated paper.**

The third experiment was designed to evaluate the accuracy of the depth estimation of the algorithm implemented in the active stereo vision platform by setting the baseline to four values (10, 20, 30 and 40 cm) to evaluate the effect of baseline length on the depth measurement. Figure 9 shows the pattern attached to a stand. The pattern was placed at various depths (40–200 cm at 20 cm intervals). The pattern was placed on various $X$-axes to justify the accuracy of the platform. The measurement was repeated 10 times for each interval, where the starting point of the slave camera was adjusted to the zero position during each measurement. The purpose of resetting the starting point of the camera to the zero position was to evaluate the performance and repeatability of the vergence control to verge on the fixation point. Consequently, the experiment was repeated 360 times to accommodate all baselines and interval conditions.

### D. PERFORMANCE COMPARISON

The experimental system was compared to the work of Zhang (2011), in which the image resolution is 200×200 pixels with a three-level pyramid (baseline: 24 cm). Note that we used the same configuration for our system to compare the algorithms.

### E. SMALL OBJECT DEPTH DETECTION

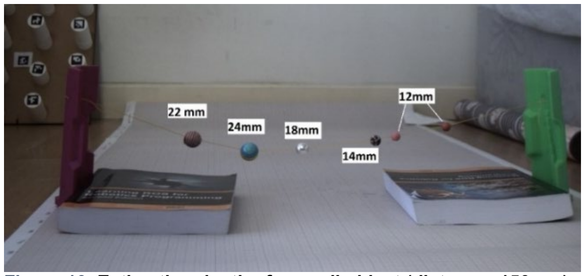


**Figure 10. Estimating depth of a small object (distance: 150 cm).**

Figure 10 shows the configuration of the fourth experiment. The main purpose of this experiment was to determine the performance of the algorithm and the system by detecting the depth of six small targets (1.2–2.4 cm). These objects were placed 150 cm away from the sensors. Then, the estimated depth was recorded.

### F. FIELD EXPERIMENTS

*Figure 11: The greenhouse experiment setup.*

Finally, the system is tested in the field with a real tomato in a greenhouse. Where the system place in front of the tomato trees (Figure 11)[1]. Two scenes were set to test the vergence controller the first one with four target at distance between 100 and 105 cm and the second scene with six targets at distance between 85cm and 95 cm that make the scene more crowded (Figure 12). Gaze on the tomato was done manually, by selecting the target in the screen then the gaze controller fixes the centroid of the target to the fixation point of the system. After the master camera fully gazed on the target the vergence controller turn on to verge on the targets. The setup of the vergence controller is full resolution image (2080×1040), the template size was set to 150×150 and the pyramid layer set to seven layers.

[1] The experiment was run in the morning on 1st of September 2018 at 10am GMT.


*Figure 12: two scenes used in testing the vergence controller and platform. (a) Four tomatoes setup (100 – 105cm) (b) Six tomatoes setup (80 – 95 cm).*

## IV. RESULTS AND DISCUSSION

### *A. PNCC RESULTS*

Here, the results of the vergence control algorithm are presented and discussed. Using the PNCC algorithm to verge the slave camera to the fixation point of the master camera has many advantages, where the correlation between both images became more robust and accurate under different lighting conditions and similar repeatable features in the scene.

The template shown in Figure 7 was used to evaluate the reliability of the algorithm. Note that the template has a pattern that can deceive the algorithm. Moreover, some template-matching algorithms, such as SAD and SSD, are sensitive to illumination changes (Gräßl el at., 2003); therefore, the experiment included changing the illumination input to the master camera by controlling the IRIS.

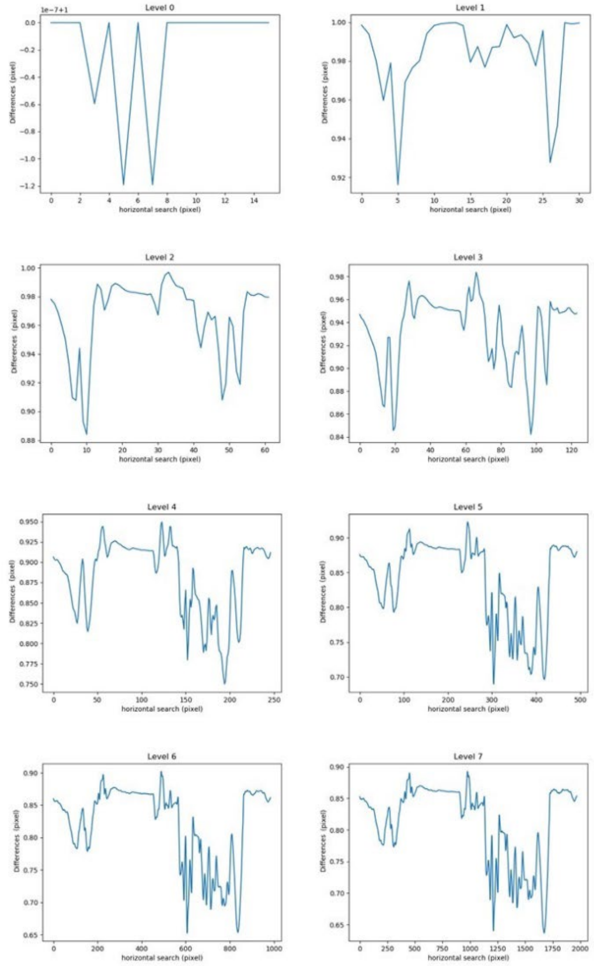


**Figure 13. Output of PNCC algorithm at target 4 in the template (target depth: 1.5 m).**

The results shown in Figure 13 are the output of the verged slave camera on target four obtained using the PNCC algorithm, where the process began at level zero and increased to level seven (image sizes of 16×8 and 2048×1080). Note that the output levels are smaller than the actual image due to spatial convolution windowing. The results show how the pyramid algorithm helps fix the slave camera's gaze by referencing the master camera's fixation point. The pyramid algorithm was used to focus the correlation on the target at coarser levels (levels 0–4) and control the precision of the output using finer levels (levels 5–7). The results are shown in Figure 14. The template has multiple targets that are similar in terms of shape, which confuses standard NCC relative to determining the target. The final verge on the fixation point has an error of ±10 pixels due to the controller's behavior. This error may occur due to the resolution of the motors, which is 12-bit (±0.088°) for pan and 10-bit (±0.29°) for tilt. Consequently, this may affect depth estimation performance slightly despite the fact that the cameras have a high resolution (4.4 MP) that help in improving the align the center of the target.

As shown in Figure 13, there are four peaks that demonstrate sharp rises through the levels, and, in level 7, these peaks are (300, 0.86), (572, 0.83), (807, 0.82) and (986, 0.98) from left to right, respectively. These peaks are the four patterns in the template with nearly equal intensity, and these patterns lie on the same horizontal line. Note that this occurs for nearly all patterns in the template. In some cases, the system verges on the pattern with high error in the final fixation point, and this is due to differences in the perspective view of the master and slave cameras at large angles. This error depends on the shape of the target, e.g., a cube target will have different shadows at different side, which consequently leads to increase the error during verge process.

Another issue that occurs when using the PNCC algorithm is the selection of the template size. The size of the template determines the precision of the target's position in the slave camera. A large template leads to a slight shift in the position due to extra neighbor pixels, which results in over-computation of the target's features. This also occurs if the template is small as there is insufficient information to consider during the search, which leads to poor vergence on the target. The target size in the image varies depending on its position in front of the camera, i.e., distant targets are smaller and vice versa. Generally, to realize accurate vergence control, the window size should be adaptive to the size of the target, which is determined by the number of the pixels. Therefore, controlling window sizes will be the focus of future research.

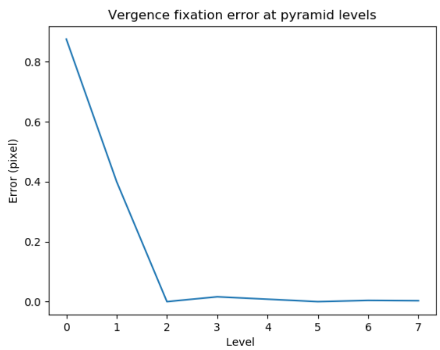


**Figure 14.** **Error in pyramid levels when the system is fully verged on the fixation point (pattern number 4).**

One disadvantage of using a pyramid in template matching is that the target search process depends on the course levels of the pyramid (i.e., the top), where, if the output location of the target at the course level is in the wrong position, the rest of the search in subsequent layers will be in this incorrect region. Typically, this occurs when testing the system in a busy environment. Furthermore, the system fails when both cameras rotate more than 35°, where the perspective of both views become large which leads to failure in vergence control. In some cases, the vergence tracking is overcome if the object is spherical.

### *B. UNBALANCE LIGHTNING CONDITIONS FOR MASTER AND SLAVE*

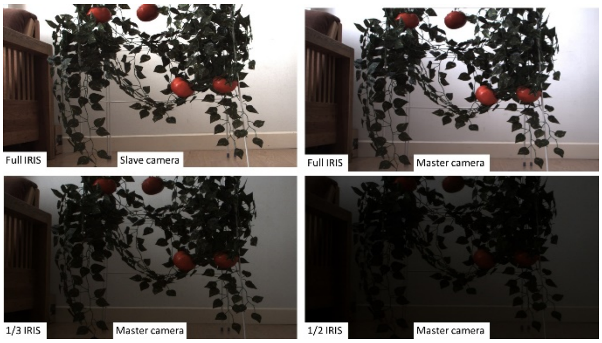


**Figure 15.** **Different Brightness between the master and slave images.**

In this experiment, the PNCC algorithm was tested against changing brightness levels between the master and slave cameras. Figure 15 shows the three lighting conditions input to the master camera (lighting conditions for the slave camera were unchanged).

This setup helps identify the robustness of the algorithm under different lighting condition, e.g., if one camera faces the light source and the other camera light source is behind the other camera, the camera facing the light source will have a darker image due to limitations in camera dynamic range.

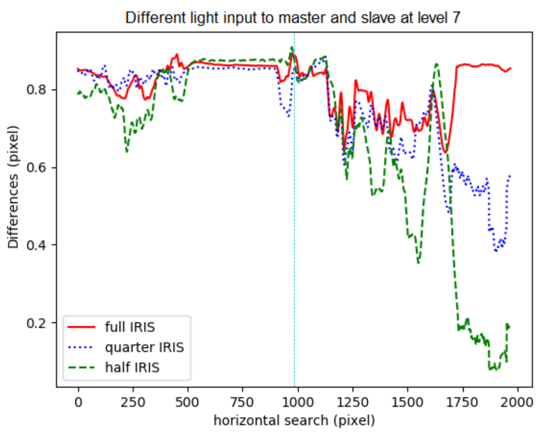


**Figure 16.** **Vergence at target under three lighting conditions by controlling camera IRIS.**

Figure 16 shows the output of level 7 when the system verged on the target under different lighting conditions for the master camera. Here, the vertical dashed line represents the center of the image (where the target should be located). The error was calculated by measuring the difference between the centers to the peak of the PNCC output. The error was 0.71%, 1.3% and 1.9% for the equal, one-quarter and one-half IRIS configurations between the master and slave cameras, respectively. These errors could be due to the arithmetic precision of the PNCC calculations ($\pm$ 1 pixel), wherein some steps are required to use an integer rather than a float. Alternatively, these errors could be due to the controller margin error ($\pm$10 pixel) that was set to stop the isolating. Considering the camera used in this work (i.e. 12-bit which has over 4,000 tones) the image processing algorithm will not lead to large error.

These results demonstrate promising performance in a practical application. The results of this experiment are significant for computer vision and robotics fields because, in practical application, many noise sources exist, and it is difficult and expensive to control all of them, especially in computer vision implementations.

### C. DEPTH ESTIMATION RESULTS

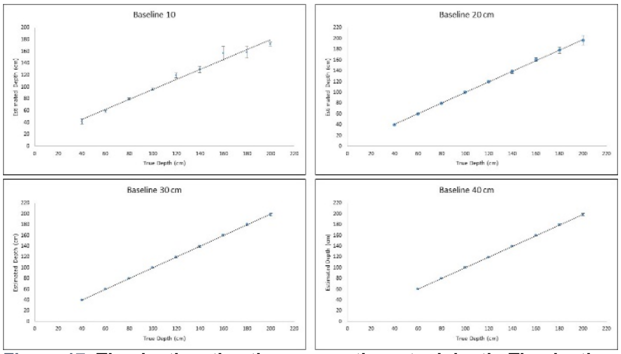


**Figure 17. The depth estimation versus the actual depth. The depth estimation was measure at four baseline distance. The result shows the error bar of the experiments at different location.**

The accuracy of the vergence depth is influenced by the accuracy of the mechanical joints and encoder resolution of the platform. A quantitative measurement was performed to evaluate the accuracy of the system. Figure 17 shows the depth measurements results with the standard deviation (error bar) for various baselines. In Section II.C, the behavior of the measurement was explained theoretically, and the experimental results validate the stated theory. The results show that the error and estimated depth are directly proportional to each other, where a rise of either of these factors results in an increase in the other one. Here, the depth is inversely proportional to the verge angle, where an increase in-depth reduces the verge angle. On the other hand, the baseline length is directly proportional to the verge angle, where any expansion in the baseline results in a larger verge angle. Overall, these observations are summarized in Figure 18. This behavior is clearly shown in Figure 17, e.g., the estimated depth of the target at 200 cm is improved dramatically by increasing the baseline (10, 20, 30, and 40 cm) because the estimated depth is 172.22 ± 3.45 cm, 196.83 ± 9.1 cm, 198.02 ± 2.7 cm, and 198.64 ± 2.06 cm, respectively. The overall results show that baseline size yields limited improvement relative to depth estimation due to the relationship between the verge angle and the depth estimation.

The large standard deviation formed in baseline 30 cm and baseline 40 cm is partially related to the controller margin error (±10 pixels) and the mechanical assembly, such as error in the position of the camera origin. In our previous work [30], error in the mechanical joints was measured and analyzed, and the overall error was ±0.01 cm. Moreover, depth estimation is related to the resolution, where finer resolution results in a more precise vergence control and finer depth measurement. For example, if the target at 100 cm and moves to a new position that is 120cm perpendicular to the baseline that the depth measure at travel of 1.5 cm. For high-resolution images, the system measures the target in millimeters (100, 101.5, 103.0, .., 120 ± 1.0 cm); however, for low-resolution images, the system measures the target in centimeters (i.e. 101, 102, .., 120 ±1 cm).

| Variables / Relation | Percentage of Error (E) | Depth (D) | Verge Angle (V) | Baseline (B) |
|---|---|---|---|---|
| $\delta E / \delta D$ | ↑ | ↑ | ↓ | — |
| $\delta D / \delta V$ | ↓ | ↓ | ↑ | — |
| $\delta V / \delta B$ | ↓ | — | ↑ | ↑ |

**Figure 18. Error relationship between verge angle, depth estimation and baseline.**

TABLE II summarizes the depth estimation results with four baselines, i.e., 10–40 cm with an increment of 10 cm. The table shows the average depth, standard deviation in centimeters and the Mean Absolute Error (MAE) in percentage. For the 10 cm and 20 cm baselines, the MAE increased sharply as the depth increased. For the 30 cm and 40 cm baselines, the MAE increased slightly with increased depth. However, for baselines greater than 30 cm, the minimum depth should be set to 60 cm because, at most positions less than 60 cm, the system failed to verge on the target due to the large perspective view between both images.

TABLE II

DEPTH ESTIMATION SUMMARY OF FOUR BASELINES

| True Depth | Baseline 10 cm | | Baseline 20 cm | | Baseline 30 cm | | Baseline 40 cm | |
|---|---|---|---|---|---|---|---|---|
| | Avg. depth ± std (cm) | MAE (%) | Avg. depth ± std (cm) | MAE (%) | Avg. depth ± std (cm) | MAE (%) | Avg. depth ± std (cm) | MAE (%) |
| 40 | 41.93 ± 4.1 | 2.36 | 39.96 ± 0.64 | 0.38 | 39.89 ± 0.37 | 0.31 | Fail | 0.00 |
| 60 | 59.5 ± 2.01 | 1.55 | 60.14 ± 0.65 | 0.48 | 60.22 ± 1.02 | 0.77 | 60.35 ± 0.89 | 0.69 |
| 80 | 79.68 ± 1.69 | 1.33 | 79.89 ± 0.82 | 0.69 | 80.02 ± 0.83 | 0.64 | 79.99 ± 0.89 | 0.55 |
| 100 | 96.47 ± 2 | 3.67 | 100.49 ± 1.96 | 1.57 | 100.19 ± 0.85 | 0.67 | 100.35 ± 0.95 | 0.71 |
| 120 | 119.37 ± 4.04 | 2.77 | 120.08 ± 1.71 | 1.43 | 119.49 ± 1.33 | 1.24 | 119.29 ± 0.76 | 0.88 |
| 140 | 129.82 ± 5.22 | 10.18 | 139.06 ± 3.51 | 3.03 | 139.43 ± 2.14 | 1.52 | 139.43 ± 1.00 | 0.84 |
| 160 | 157.07 ± 11.57 | 8.24 | 160.66 ± 3.55 | 2.76 | 160.39 ± 1.96 | 1.50 | 159.99 ± 1.06 | 0.86 |
| 180 | 158.99 ± 9.99 | 21.00 | 178.66 ± 5.56 | 4.98 | 180.21 ± 2.12 | 1.73 | 179.56 ± 2.14 | 1.60 |
| 200 | 172.22 ± 3.45 | 27.77 | 196.83 ± 9.1 | 8.10 | 198.02 ± 2.7 | 2.98 | 198.64 ± 2.06 | 2.12 |

Another depth estimation experiment was performed with the tomato setup in the lab (Figure 8) to evaluate the reliability and repeatability of tracking a target in a complex scene. Here, the master camera was fixed on the target manually, and the slave camera had a different starting point. The baseline was set to 20 cm, and the measurement was repeated 10 times for each target.

TABLE III

DEPTH ESTIMATION WITH LAB TOMATO SETUP (BASELINE: 20 CM)

| Target | True depth (cm) | Avg ± std (cm) | MAE (%) |
|---|---|---|---|
| 1 | 150 ± 0.2 | 148.91 ± 2.13 | 2.15 |
| 2 | 140 ± 0.2 | 139.81 ± 1.25 | 1.11 |
| 3 | 135 ± 0.2 | 132.24 ± 1.73 | 2.76 |
| 4 | 126 ± 0.2 | 125.97 ± 0.86 | 0.68 |

Table III shows the experimental results of the lab setup of the artificial tomato. As can be seen, the results are nearly the same as the results of the depth estimation experiment. This experiment illustrates how the platform can perform under different targets and different backgrounds. Moreover, in this experiment, the target was fixed and the vergence controller start at different starting position; therefore, the results demonstrate the reliability (almost equal $\pm$ 3 cm) of the PNCC algorithm using the propose platform. Figure 19 shows the output of the vergence controller when it fully verge on the targets.


**Figure 19. The result of the vergence controller verged on two different target (A) and (B) in the lab setup.**

Thus, we conclude that the overall performance of the system depends on the size of the baseline. When the baseline is less than 20 cm, the error in-depth estimation increases, leading to poor results. However, when the baseline is set to a value greater than 20 cm, depth estimation is improved to up to $\pm$2.1 cm at a depth of 200 cm.

## D. COMPARISON OF EXPERIMENTAL PLATFORM TO PREVIOUS WORK

Table IV compares the depth estimation outputs of our system and Zhang's system. The system configuration, such as baseline, image size and pyramid levels, was set according to Zhang's system (baseline: 24 cm, image size: 200×200, three pyramid levels). The results show that our system demonstrates slightly better improvement in terms of depth estimation because the average absolute error is better. In terms of standard deviation, our system outperformed Zhang's system in all three runs, where our system achieved an average error of $\pm$1.83% compared to $\pm$3% for Zhang's system. It is apparent that Zhang's system shows a large standard deviation when it comes to measuring depths closer to the system. As a result, distortion of the perspective view of the object increases in the log-polar space.

TABLE IV
DEPTH ESTIMATION RESULTS OF ZHANG'S AND PROPOSED SYSTEMS

| True Depth (cm) | Zhang's system | | Our system | | Average error $\pm$ std dev differences (%) |
|---|---|---|---|---|---|
| | Average estimated depth (cm) | Average error $\pm$ std dev (%) | Average estimated depth (cm) | Average error $\pm$ std dev (%) | |
| 80 | 84 | $5 \pm 2$ | 80.66 | 1.13 $\pm$ 1.74 | 3.87 $\pm$ 0.26 |
| 100 | 106 | $6 \pm 3$ | 101.17 | 1.57 $\pm$ 1.73 | 4.43 $\pm$ 1.27 |
| 180 | 182 | $3 \pm 3$ | 179.87 | 2.78 $\pm$ 1.83 | 0.22 $\pm$ 1.17 |

## E. SMALL OBJECT DEPTH ESTIMATION

In this experiment, the depth of small objects was measured using the three sensors (Figure 10). TABLE V shows the depth measurements. Here, our system detected all targets. The output of our system shows that the MAE and std reduced as the diameter of the objects increased, where the MAE and std values dropped from $\pm$2.58% to $\pm$1.06% and $\pm$2.49 cm to $\pm$1.12 cm, respectively. This variation was due to the size of the windows and controller error. Our system estimated the depth of all targets because the search process was performed for one target at a time, which depends on the features of the target directly from the image rather than computing the disparity of the entire scene, as is the case with the orthogonal stereo vision systems.

TABLE V
RESULTS OF SMALL OBJECT DEPTH ESTIMATION EXPERIMENT

| No. | Target Diameter (cm) | | |
|---|---|---|---|
| | | Avg ± std (cm) | Mean Absolute Error (%) |
| 1 | 1.2 | 152.53± 2.49 | ±2.58 |
| 2 | 1.2 | 152.85± 2.42 | ±2.86 |
| 3 | 1.4 | 151.84± 1.84 | ±2.15 |
| 4 | 1.8 | 150.47± 2.19 | ±1.84 |
| 5 | 2.2 | 151.12± 2.24 | ±1.96 |
| 6 | 2.4 | 150.74± 1.12 | ±1.06 |

## F. FIELD EXPERIMENT RESULTS

Two types of results are evaluated in this experiment: (I) the depth estimation of each target and (II) the verge on the target by computing the disparity.

*Figure 20: the output of vergence controller on scene 1.*

*Figure 21: the output of vergence controller on scene 2.*

Figure 20-Figure 21 shows the output of the vergence controller experiment when the system verge on the fixation point. Figure 20 shows scene one where there are four tomato fruits while Figure 21 shows scene two with six tomato fruits. The difference between scene one and scene two is that scene two contains more targets that are close to each other which challenge the system to verge correctly on the fixation point of the gaze. In both the scenes, the system manages to verge on the target with an accuracy of $\pm 10$ pixels. This range is due to the control margin set in the controller to stop the isolation and also supported by the lab evaluation.

The vergence controller performs reliably well when the master camera fully verges on the target and stay still, but when the master camera moves from one target to another, the vergence controller lose the tracking of the master camera particularly when the master camera on the leafs. The reason is that the background shares a common intensity value. However, this is not a big issue as long as the vergence controller verged correctly on the final fixation point, or this could be avoided by suspending the vergence controller when the master camera changes the targets.

In the second evaluation of the depth estimation, TABLE VI shows the result. The platform was set at a baseline of 20 cm, and the targets were between 85cm and 105cm. The depth estimation in the greenhouse has larger std value compared to the lab result, which is due to the noise present in outdoor environments. For example, for the target at scene two, the std is $\pm 1.32$ cm at a depth of 85cm which is 1.5 times bigger than the result of the lab.

TABLE VI DEPTH ESTIMATION FOR FIELD EXPERIMENTS

| Scene 1 | | |
|---|---|---|
| Target | True depth (cm) | Avg ± std (cm) |
| a | 96.8 | ±1.02 |
| b | 96 | ±0.71 |
| c | 101.8 | ±0.84 |
| d | 104.6 | ±1.14 |
| Scene 2 | | |
| Target | True depth (cm) | Avg ± std (cm) |
| a | 85.6 | ±1.32 |
| b | 91.2 | ±1.30 |
| c | 86.8 | ±1.30 |
| d | 94.8 | ±0.45 |
| e | 85 | ±0.71 |

## V. CONCLUSION

In this paper, the PNCC algorithm for vergence control has been studied. The proposed model was integrated with a binocular platform that has five degrees of freedom. The model was designed to overcome the problems in the traditional NCC algorithm by introducing a Gaussian pyramid method. Improvements were observed with respect to the accuracy and reliability of the controller and stability of fixation on the target. The verge on the target in a complex environment and presence of similar shapes of the proposed algorithm is more robust than conventional NCC and other template-matching techniques where the maximum difference is $\pm 10$ pixels. Our experimental system demonstrates good vergence on the target under different lighting conditions between the master and slave cameras while maintaining the same output when both cameras have the same IRIS configuration.

Through multiple quantitative and qualitative experiments, the experimental system demonstrated good depth estimation with standard deviation of $\pm 2.06$ cm at a worst-case depth of 200 cm, where the depth estimation improved when the baseline was greater than 20 cm MAE equal to 2.21% (TABLE II). In addition, the system showed improvement compared to an existing method in terms of depth estimation and reliability, where the overall highest percentage of error at three different depths was 4.43% at 100 cm. The experimental system was also compared to other stereo vision sensors, and better depth estimation results were obtained despite a minor disadvantage of the system, i.e., it can only obtain the depth of one target at a time, thereby making it slower than orthogonal stereo vision systems.

The experiments conducted showed that the system requires certain improvements to ensure that the cameras perform more precisely when rotates more than 35 degrees. Therefore, a future research question can be posed which is how to minimize the perspective distortion error.

However, another experiment was conducted to test the capability of estimating the depth of a small objects (e.g., 1.2 and 1.4 cm objects), which are difficult to estimate with most stereo vision systems. The results indicate that our experimental system demonstrates reliable and robust depth estimation for such targets at a standard deviation of $\pm 1.12$ cm at 150 cm.

The algorithm and the platform was tested in outdoor environment with a tomato bush. The result shows that the system operates outdoors, in natural lighting, with a std $\pm 1.32$ cm at 85 cm with a verge error of $\pm 10$ pixels.

In future, the proposed PNCC algorithm could be improved by implementing an adaptive template size, which will allow vergence on the fixation point to be more precise when it changes based on the target's size and position. In addition, the experimental system will be tested in a practical environment for estimating the position of tomatoes fruit harvesting process.

## ACKNOWLEDGMENT

The author would like to thank Jaafar Mohamed for his support and proofreading.

## REFERENCES

[1] J. C. A. Read, "What is stereoscopic vision good for?" vol. 9391, p. 93910N, 2015.
[2] V. Nityananda, G. Tarawneh, R. Rosner, J. Nicolas, S. Crichton, and J. Read, "Insect stereopsis demonstrated using a 3D insect cinema.," *Scientific Reports*, vol. 6, p. 18718, 2016.
[3] J. R. Jensen, "Remote sensing of the environment: an earth resource perspective," 2nd ed., NJ: Pearson Prentice Hall, 2007.
[4] S.-W. Bana and M. Lee, "Biologically Motivated Vergence Control System Based on Stereo Saliency Map Model," in *Scene Reconstruction Pose Estimation and Tracking*, I-Tech Education and Publishing, 2007.
[5] B. Rogers and M. Graham, "Similarities between motion parallax and stereopsis in human depth perception," *Vision Research*, vol. 22, no. 2, pp. 261–270, 1982.
[6] K. Kanatani, "Statistical Optimization for Geometric Computation: Theory and Practice," 1st ed., New York: Dover Publications, 2005.
[7] T. Dang, C. Hoffmann, and C. Stiller, "Continuous stereo self-calibration by camera parameter tracking," *IEEE Trans. Image. Process*, vol. 18, no. 7, pp. 1536–1550, Jul. 2009.
[8] Y. Xiong and L. Matthies, "Error analysis of a real-time stereo system," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pp. 1087–1093.
[9] N. Sabater, J.-M. Morel, and A. Almansa, "How Accurate Can Block Matches Be in Stereo Vision?," *SIAM J. Imaging Sci.*, vol. 4, no. 1, pp. 472–500, Jan. 2011.
[10] A. Mohamed, P. F. Culverhouse, R. De Azambuja, A. Cangelosi, and C. Yang, "Automating Active Stereo Vision Calibration Process with Cobots," *IFAC-PapersOnLine*, vol. 50, no. 2, pp. 163–168, Dec. 2017.
[11] A. Gibaldi, M. Vanegas, A. Canessa, and S. P. Sabatini, "A Portable Bio-Inspired Architecture for Efficient Robotic Vergence Control," *Int J Comput Vis*, vol. 121, no. 2, pp. 281–302, Jan. 2017.
[12] Jian Peng, A. Srikaew, M. Wilkes, K. Kawamura, and A. Peters, "An active vision system for mobile robots," in *SMC 2000 Conference Proceedings. 2000 IEEE International Conference on Systems, Man and Cybernetics. "Cybernetics Evolving to Systems, Humans, Organizations, and their Complex Interactions" (Cat. No.00CH37166)*, 2000, vol. 2, pp. 1472–1477.
[13] P. Culverhouse *et al.*, "Vision Processing on the Bunny Robot Humanoid Robot," in *In Proceedings of the 4th workshop on humanoid soccer robots a workshop of the 2009 IEEE-RAS international conference on humanoid robots (Humanoids 2009)*, 2009, pp. 60–65.
[14] P. Von Kaenel, C. Brown, and D. Coombs, "Detecting Regions of Zero Disparity in Binocular Images," 1991.
[15] M. M. Marefat, L. Wu, and C. C. Yang, "Gaze stabilization in active vision—I. Vergence error extraction," *Pattern Recognition*, vol. 30, no. 11, pp. 1829–1842, Nov. 1997.
[16] A. Dankers, N. Barnes, and A. Zelinsky, "MAP ZDF segmentation and tracking using active stereo vision: Hand tracking case study," *Comput Vis. Image Underst.*, vol. 108, no. 1–2, pp. 74–86, Oct. 2007.
[17] X. Zhang and L. P. Tay, "A spatial variant approach for vergence control in complex scenes," *Image Vis. Comput.*, vol. 29, no. 1, pp. 64–77, Jan. 2011.
[18] A. Bernardino and J. Santos-Victor, "Vergence control for robotic heads using log-polar images," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS '96*, 1996, vol. 3, pp. 1264–1271.
[19] A. X. J. Zhang, A. L. P. Tay, and A. Saxena, "Vergence Control of 2 DOF Pan-Tilt Binocular Cameras using a Log-Polar Representation of the Visual Cortex," in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, 2006, pp. 4277–4283.
[20] C. Capurro, F. Panerai, and G. Sandini, "Dynamic vergence," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS '96*, 1996, vol. 3, pp. 1241–1248.
[21] S. Rougeaux, N. Kita, Y. Kuniyoshi, S. Sakane, and A. S. Section, "Tracking A Moving Object With A Stereo Camera Head," *n Proc. 11th Annual Conf. of Robotics Society of Japan*, pp. 1–4, 1993.
[22] B. Cyganek and J. P. Siebert, "An introduction to 3D computer vision techniques and algorithms." John Wiley & Sons, 2009.
[23] C. Yim and A. C. Bovik, "Using a Hierarchical Image Structure Pb ' rbl," *Control*, pp. 0–5, 1994.
[24] R. Hartley and A. Zisserman, "Multiple view geometry in computer vision," 2nd ed., Cambridge: cambridge university press, 2003.
[25] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, "Automatic generation and detection of highly

reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280–2292, 2014.

[26] G. R. Bradski and A. Kaehler, *Learning OpenCV: computer vision with the OpenCV library*. O'Reilly, 2008.

[27] Y. Fouda and K. Ragab, "An efficient implementation of normalized cross-correlation image matching based on pyramid," in *2013 International Joint Conference on Awareness Science and Technology & Ubi-Media Computing (iCAST 2013 & UMEDIA 2013)*, 2013, pp. 98–103.

[28] H. Sahabi and A. Basu, "Analysis of error in depth perception with vergence and spatially varying sensing," *Comput. Vis. Image Underst.*, vol. 63, no. 3, pp. 447–461, 1996.

[29] C. Gräßl, T. Zinßer, and H. Niemann, "Illumination Insensitive Template Matching with Hyperplanes," in *Pattern Recognition*, 2003, pp. 273–280.

[30] A. Mohamed, P. F. Culverhouse, A. Cangelosi, and C. Yang, "Active Stereo Platform: Online Epipolar Geometry Update," *EURASIP J. Image Video Process.*, 2018