Accuracy Amelioration of an Integrated Real-Time 3D Image Sensor

Jad Ayoub, Olivier Romain, Bertrand Granado, Yasser Mohanna

Accuracy Amelioration of an Integrated Real-time 3D Image Sensor

J. Ayoub[1], O. Romain[1], B. Granado[2], and Y. Mhanna[3]

Abstract— In this paper we investigate an active vision technique implemented in an embedded system for 3D shapes reconstruction. The main objective of the work is to have a balance in the accuracy of all components in the system where the size and autonomy of such an embedded sensor are hard constraints. This is achieved through the improvement of the pre-processing algorithms by reducing the time needed to compute the spots centers. In addition, lens distortion of the camera is included in the model to increase accuracy when reconstructing objects. Experimental evaluation shows that the size and the time are reduced, precision increased, when the resources spent on processing are relatively acceptable in comparison to the benefits.

Index Terms— **3D reconstruction, Active stereovision, Camera calibration, Lens distortion.**

I. INTRODUCTION

RECONSTRUCTING 3D shapes is needed in several applications in computer vision and computer graphics, namely object recognition for robotic vision.

Numerous techniques have been developed to give solutions to the 3D reconstruction problem. The most common are those based on vision systems basing on either passive or active stereovision methods, where image sensors are used to provide the necessary information to retrieve the depth, since is not the case in traditional photography. The most commonly employed passive method consists of taking two images of a scene at two different shooting angles using either two cameras or only one camera for which an acquisition in two different positions is done. Then the 3-D coordinates of any point can be deduced from the 2-D coordinates by triangulation. Using this method, only characteristic points, with high gradient or high texture can be detected [1].

The active stereovision methods offer an alternative approach to the use of two cameras. They consist in replacing one of the two cameras by a projection system which projects a set of structured rays. In this case, only one image is necessary. Many implementations of active stereovision methods have been realized. Some of them [2,9] provided significant results using traditional computer for application of such methods. In our research work, we have focused on an integrated 3D active vision sensor: "Cyclope" [1]. This sensor allows making real time 3D reconstruction while respecting the size and power consumption constraints of embedded systems [12] to be used in special applications like wireless capsule endoscopes, robotic heart surgery, or even asteroids exploration [11].

To realize this sensor, many techniques should be involved, starting with image capturing, spatial filtering, morphological operations, conversion gray-scale image to binary, segmentation, region labeling, correction of lens distortion, computation of spot centers, matching centers to epipolar lines, and finally 3D shape reconstruction. All these techniques will be realized in real-time. Beside, other necessary techniques should be executed off-line, like calibration of camera lens distortion, determination of epipolar lines and depth model.

Among the steps listed above, we will focus in this paper on two influencing parts to improve the accuracy of our sensor: the computation method of the spots centers, and the correction of lens distortion that highly affects the resulting measures.

The second section describes briefly Cyclope. Section 3 deals with the principles of the active stereovision system and 3D reconstruction method, explaining problem statement. In section 4 we discuss lens distortion and its influence on hardware implementation. In section 5, we present methods used to extract the centers of laser spots taking into account time and size demands. In section 6 we summarize the experimental results. And finally we conclude in section 7.

II. CYCLOPE

Cyclope is our integrated wireless 3D vision system based on active stereovision technique, it uses many different algorithms to increase accuracy and reduce processing time and sensor size. Such properties let this sensor more compatible for emergent and demanding applications where size and autonomy are hard constraints. The block diagram of "Cyclope" (figure 1) is composed of three essential parts [1]:

[1] J. Ayoub, O. Romain, Université Pierre et Marie Curie (UPMC), groupe SYEL, 4 place Jussieu, Paris, France (jad.ayoub@etu.upmc.fr; romain@upmc.fr)

[2] B . Granado, ETIS, CNRS, ENSEA, Université Cergy Pontoise 95014 CERGY cedex, France, (bertrand.granado@ensea.fr)

[3] Y. Mhanna, Department of Physics and Electronics, Faculty of Sciences I, Lebanese University, Beirut, Lebanon (yamoha@ul.edu.lb)

- Instrumentation block: containing CMOS camera and a structure light projection system.
- Processing block: integrates a microprocessor core and a reconfigurable array. The microprocessor is used for sequential processing and the reconfigurable array is used to implement time consuming algorithms.
- Wireless communication block: This part is dedicated to the OTA (Over the Air) communication to have a wireless sensor.
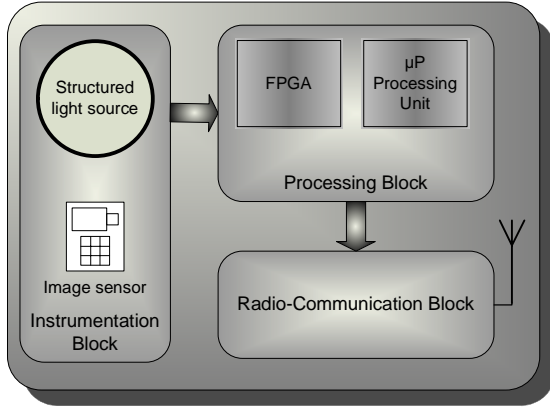

Fig. 1. Block diagram of CYCLOPE

To test and validate our system, a large scale demonstrator have been realized using an original CMOS imager, a generator of structured light constituted by an array of 361 (19x19 laser each being separated from its neighbors by a fix angle equal to 0.77°), XUP Virtex-II Pro Development System Board, and a Zigbee module.

III. ACTIVE STEREOVISION SYSTEM AND 3D RECONSTRUCTION

A. Principle and mathematical model

Active stereovision system consists of a single camera and a generator of laser-structured light that replaces the second camera of the passive stereovision system [1].

The laser projector is combined with a diffraction network in order to illuminate the studied scene with an array of laser beams. Each ray is separated from its neighbors by a fix and equal angle. The setup of active stereovision system is
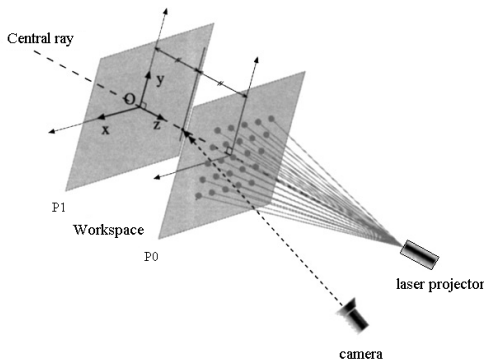

Fig. 2. Active stereovision system

represented in figure 2.

The calibration is processed according to a workspace delimited by two planes P_0 and P_1 perpendicular to the central ray of the beam, within which the object to be analyzed is placed [3].

The 3D reconstruction is achieved through triangulation between laser and camera. Each point of the projected pattern on the scene represents the intersection of two lines:

- The line of sight, passing through the pattern point on the scene and its projection in the image plane.
- The laser ray, starting from the projection center and passing through the chosen pattern point.

The calibration process provides two sets of parameters for each of the 361 points of the mesh [3]:

- A set of segments (eq.1), each representing the projection, in the image plane, of the part of a ray which crosses the workspace. This projection is the epipolar line in a passive stereovision system with two cameras where the laser ray is identified to the second camera line of sight (see Fig. 4(b)).
- The relationship between the center position of a given point on the corresponding segments of the ray and depth of the physical point corresponding to the point. This relation is modeled through an hyperbolic curve (eq.2).

$$v=a.u+b \quad \text{with} \ (a,b) \in \Re^2 \quad \text{(eq.1)}$$

$$z=\frac{1}{\alpha u + \beta} \quad \text{with} \ (\alpha,\beta) \in \Re^2 \quad \text{(eq.2)}$$

(u,v) are the image coordinate of a laser point.

B. Calibration

The values of parameters of equations (1) and (2) are obtained off-line using a least-square fit on the experimental data obtained through a calibration process. Note that the data are calculated considering lens distortion correction of the camera. Figure 4(a) represents the sequence of 12 planes used for calibration.

The calibration process yields a set of segments that represents the projections of the laser rays on the image plane (see Figure 4(b)). Our goal is to determine, for each spot appearing on the image, the segment on which it lies and, consequently, to associate the spot with the ray from which it stems [3].

C. Workflow of 3D reconstruction process

As shown in Figure 3, the scene illuminated by structured light source is captured by a CMOS camera and then stored into memory. The captured image submits a pre-processing phase to extract the useful data concerning the light spots.

The first stage of the pre-processing phase is to apply a median filter to blur the boundaries between bright spots and dark background and to remove outlier pixel intensities. The next stage is to apply a thresholding operation in order to
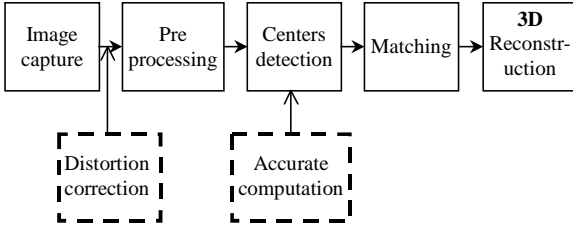
Fig. 3. A flowchart showing different phases of 3D image reconstruction

classify the bright areas and the dark area. The threshold process often produces an image that is less than perfect and is not sufficient to separate objects from their background, common problems are noise produced by incoherent lighting. It is often desirable to process a binary image before analysis to remove these abnormalities. This accomplished by applying morphological operations (erosion) on the binary picture to remove pepper noise remaining from the median filtration process that was applied in the first stage. Then a segmentation process is applied to extract and label the foreground subjects from the scene, to compute later their centers positions.

After that, the spots centers have to be matched to corresponding epipolar lines that were obtained off-line
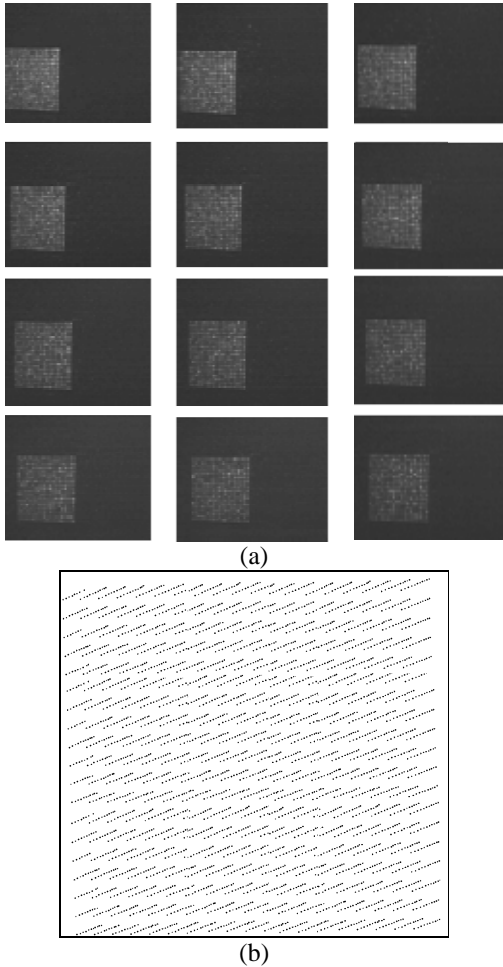

(a)

(b)

Fig. 4. (a) Calibration process: workspace delimited between 42cm and 64cm. (b) Epipolar lines

through a calibration process [2].

At the end, the distance between each spot and the stereoscopic system is computed from the depth model that is also obtained off-line through calibration process [2].

This workflow is represented in Figure 3 by blocks with solid lines. Since, the blocks with dashed lines represent the improvement implemented in our work to ameliorate the accuracy of such a system.

D. Problem statement

As equations (1) and (2) show, the 3D reconstruction of an object depends on the centers coordinates *(u,v)* of light spots in the image plane. Thus, any inaccurate representation of these points will highly affect the accuracy of our results in the 3D estimation stage. Indeed, many problems were encountered when performing this procedure, in this paper we will focus on two important aspects:

- Lens distortion prevents accurate perception of range [5], just because the true coordinates of laser spots are deviated due to lens distortion. That makes measurement and distance judgment difficult. So, distortion correction process will be necessary to accurately reconstruct the 3D coordinates of studied object. In our work, we will focus on correcting distortion taking into account the size, time, and power consumption constraints of embedded system.
- How to compute the coordinate of spot centers without consuming a considerable amount of resources, and considering high accuracy of our embedded system?

IV. DISTORTION CORRECTION

A. Pinhole camera model:

Physical camera parameters are commonly divided into extrinsic and intrinsic parameters [14]. Extrinsic parameters define the location and orientation of the camera reference frame with respect to the world coordinate system. Whereas the intrinsic parameters are used to link the pixel coordinates of an object with corresponding world coordinates in the
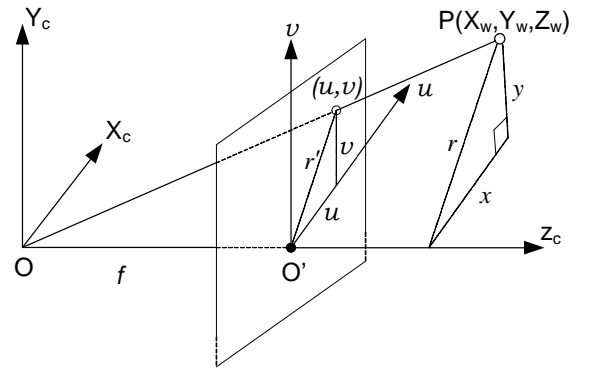

Fig. 5. Pinhole camera model
(X_w,Y_w,Z_w): World coordinates, (O,X_c,Y_c,Z_c): camera coordinates, (O',u,v): image plane coordinates.

camera reference frame, they characterize the optical, geometric, and digital characteristics of the camera.

As a first approximation, we will consider our model as a pinhole camera (see fig. 5) that neglects all optical distortion.

The relationship between world coordinate (X_w, Y_w, Z_w) and camera coordinates (X_c, Y_c, Z_c) defines the image extrinsic parameters [13], and is expressed in expression (3):

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + T \quad (3)$$

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \text{ and } T = \begin{bmatrix} t_1 & t_2 & t_3 \end{bmatrix}^t$$

Where R is a 3x3 rotation matrix defining camera orientation, and T is a 3x1 translation vector representing the distance from the origin of the camera coordinate system to the origin of the world coordinate system.

Since, the intrinsic parameters are obtained by using simple triangulation for the pinhole model; the projection of the point P to the image plane is expressed in (4), where f is the focal length:

$$\begin{bmatrix} \tilde{u}_i \\ \tilde{v}_i \end{bmatrix} = \frac{f}{Z_c} \begin{bmatrix} X_c \\ Y_c \end{bmatrix} \quad (4)$$

Where $(\tilde{u}_i, \tilde{v}_i)$ represents true coordinates (undistorted).

the relationship between the image pixel coordinates and the image coordinates is given by the expressions:

$$\begin{bmatrix} u'_i \\ v'_i \end{bmatrix} = \begin{bmatrix} D_u s_u \tilde{u}_i \\ D_v \tilde{v}_i \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (5)$$

Where (u_0, v_0) are the coordinates of the principal point O', (D_u, D_v) are conversion factors that change metric units to pixels, and s_u is the scale factor [13].

B. Lens Distortion

Pinhole camera model is based on the principle of co-linearity where each point in the object space is projected by a straight line through the projection center into the image plane. This model can be used only as an approximation of the real camera that is actually not perfect and sustains a variety of aberration [13]. So, pinhole model is not valid when high accuracy is required like in our expected applications (Endoscopes, robotic surgery..). In this case, a more comprehensive camera model must be used, taking into account the corrections for the systematically distorted image coordinates.

As a result of several types of imperfections in the design and assembly of lenses composing the camera optical system, the real projection of the point P in the image plane expressed above in expressions (5) , will be replaced by expressions that take into account the error between the real image observed coordinates and the corresponding ideal (non observable) image coordinates.

$$\begin{cases} u'=u+\delta_u(u,v) \\ v'=v+\delta_v(u,v) \end{cases} \quad (6)$$

Where (u,v) are the ideal non-observable, distortion-free image coordinates, and (u',v') are the corresponding real coordinates, δu and δv are respectively the distortion along the u and v axes.

Usually, the lens distortion consists of radial symmetric distortion, decentering distortion, affinity distortion, and non-orthogonality deformations.
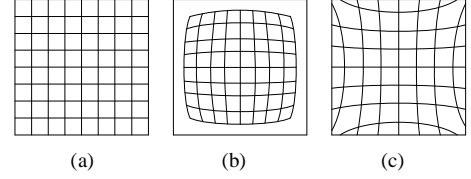

Fig. 6. : (a) The ideal undistorted grid. (b) Barrel distortion.
(c) Pincushion distortion

i) Radial distortion: is caused by flawed radial curvature of a lens and causes the actual image point to be displaced radially in the image plane. A negative radial displacement of the image points is referred to as barrel distortion, and due to the fact that many wide angle lenses have higher magnification in the image center than at periphery. This causes the image edges to shrink around the center and form a shape of a barrel. The pincushion distortion is the inverse effect when the edges are magnified stronger.

As illustrated in Figure 6, this type of distortion is strictly symmetric about the optical axis, and can be approximated [15] using the expression (7) in terms of the Cartesian coordinates (u,v):

$$\begin{aligned} \delta_{ur} &= k_1 u(u^2+v^2)+O[(u,v)^5] \\ \delta_{vr} &= k_1 v(u^2+v^2)+O[(u,v)^5] \end{aligned} \quad (7)$$

k_1 is the coefficient of radial distortion.

ii) Decentering distortion: This type of distortion caused from the fact that the optical centers of lens surfaces are not strictly collinear. This distortion has both radial and tangential components [16], which can be written as resulting along the u and v axes in the following form [14]:

$$\begin{aligned} \delta_{ud} &= p_1(3u^2+v^2)+2p_2 uv+O[(u,v)^4] \\ \delta_{vd} &= 2p_1 uv+p_2(u^2+3v^2)+O[(u,v)^4] \end{aligned} \quad (8)$$

p_1 and p_2 are coefficients for decentering distortion.

iii) Thin prism distortion: It arises from imperfection in lens design and manufacturing, as well as camera assembly. This type of distortion can be modeled by the adjunction of a thin prism to the optical system, causing additional amounts of radial and tangential distortions [17]. Such distortions can be expressed [14] with distortion coefficients s_1 and s_2 along the u and v axes as:

$$\delta_{up}=s_1(u^2+v^2)+O[(u,v)^4]$$
$$\delta_{vp}=s_2(u^2+v^2)+O[(u,v)^4] \qquad (9)$$

Other distortion types have also been proposed in the literature [13], in most cases the error is small and the distortion component is insignificant. However, it is impossible and unnecessary to consider into account all types of distortion. Only the major three types listed above need to be considered in practice.

iv)Total distortion:
The effective distortion can be modeled by addition of the corresponding expressions. Combining (7, 8, and 9), gives the total amount of distortion along the u and v axes:

$$\delta_u(u,v)= \delta_{ur}+\delta_{ud}+\delta_{up}$$
$$\delta_v(u,v)= \delta_{vr}+\delta_{vd}+\delta_{vp} \qquad (10)$$

Assuming that only the first and second order terms are enough to compensate for the distortion, and the terms of order higher than three are negligible, we obtain [18] a camera model to become fifth order polynomials (expression 11), where (u_i,v_i) are the distorted image coordinates in pixels, and $(\tilde{u_i},\tilde{v_i})$ are true coordinates (undistorted).

$$u_i=D_u s_u(k_2\tilde{u_i}^5+2k_2\tilde{u_i}^3\tilde{v_i}^2+k_2\tilde{u_i}\tilde{v_i}^4+k_1\tilde{u_i}^3+k_1\tilde{u_i}\tilde{v_i}^2$$
$$+3p_2\tilde{u_i}^2+2p_1\tilde{u_i}\tilde{v_i}+p_2\tilde{v_i}^2+\tilde{u_i})+u_0$$
$$v_i=D_v(k_2\tilde{u_i}^4\tilde{v_i}+2k_2\tilde{u_i}^2\tilde{v_i}^3+k_2\tilde{v_i}^5+k_1\tilde{u_i}^2\tilde{v_i}+k_1\tilde{v_i}^3 \qquad (11)$$
$$+p_1\tilde{u_i}^2+2p_2\tilde{u_i}\tilde{v_i}+3p_1\tilde{v_i}^2+\tilde{v_i})+v_0$$

The inverse model is deducted in expression (12):

$$\tilde{u_i}=\frac{\tilde{u_i}'+\tilde{u_i}(a_1r_i^2+a_2r_i^4)+2a_3\tilde{u_i}'\tilde{v_i}'+a_4(r_i^2+2\tilde{u_i}'^2)}{(a_5r_i^2+a_6\tilde{u_i}'+a_7\tilde{v_i}'+a_8)r_i^2+1}$$
$$\tilde{v_i}=\frac{\tilde{v_i}'+\tilde{v_i}'(a_1r_i^2+a_2r_i^4)+2a_4\tilde{u_i}'\tilde{v_i}'+a_3(r_i^2+2\tilde{v_i}'^2)}{(a_5r_i^2+a_6\tilde{u_i}'+a_7\tilde{v_i}'+a_8)r_i^2+1} \qquad (12)$$

Where:

$$\tilde{u_i}'=(u_i-u_0)/(D_u s_u), \tilde{v_i}'=(v_i-v_0)/(D_v), \text{ and } r_i^2=\sqrt{\tilde{u_i}^2+\tilde{v_i}^2}$$

The unknown parameters $a_1,..,a_8$ are solved using direct least mean squares fitting [18] in the off-line calibration process.

C. Geometric camera calibration
The objective of the geometric camera calibration procedure is to determine the intrinsic and extrinsic parameters of the camera model [13]. There are many

proposed methods that can be used to estimate these parameters, and there are also methods that produce only a subset of the parameter estimates. We chose a traditional calibration method based on observing a planar checkerboard in front of our system at different poses and positions (see Figure 7) to solve the equations of unknown parameters (12). Some Matlab toolbox are available to perform this calibration procedure [20]. The results of the calibration procedure are presented in Table 1.
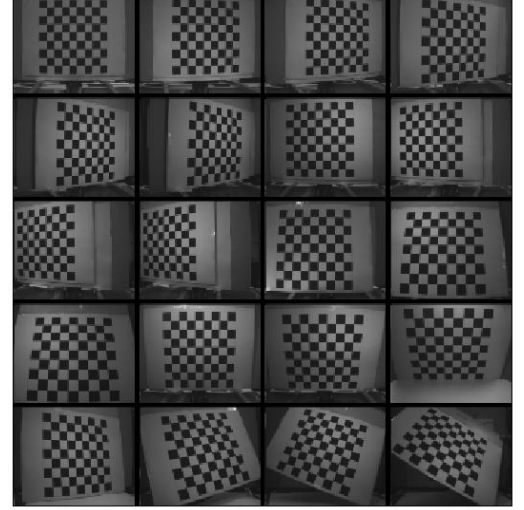

Fig. 7. Different checkerboard positions used for calibration procedure

TABLE. 1 CALIBRATION RESULTS

Parameter		Value	error
u_0	[pixels]	178.04	1.28
v_0	[pixels]	144.25	1.34
$f.D_u.s_u$	[pixels]	444.99	1.21
$f.D_v$	[pixels]	486.39	1.37
a_1		-0.3091	0.0098
a_2		-0.0033	0.0031
a_3		0.0004	0.0001
a_4		0.0014	0.0004
a_5		0.0021	0.0002
a_6		0.0002	0.0001
a_7		0.0024	0.0005
a_8		0.0011	0.0002

It is clear from the results that the two first parameters a_1 and a_2 are the dominant parameters, this fact is expected because they belong to the radial distortion, known in the literature [13, 14, 15] as the dominant distortion model. And for this reason some proposed correction methods simplify their models by restricting the lens distortion effect to radial distortion [5, 16].

In this section we have introduced the distortion correction model that we applied to both off-line calibration process needed to estimate the eppipolar lines, and to section dealing

with spots centers computation. To reduce the computation time, we have implemented the correction model only to the active light spots (361 points), where the remaining part of the picture belong to background. This improvement will increase the accuracy of our system, specially in the last stage of our work, where the essential feature of our sensor, "Cyclope", is to compute the 3D coordinates of the illuminated spots, characterized by their centers. The computation methods of these centers will be the object of the next section.

V. COMPUTATION OF SPOTS CENTERS

The Threshold and Labeling processes applied to the captured image allow us to determine the area of each spot (number of pixels). So, the centers coordinates of these points could be calculated as follow:

$$u_{gI} = \frac{\sum_{i \in I} u_i}{N_I} \quad \text{(eq. 13)} \quad \text{and} \quad v_{gI} = \frac{\sum_{i \in I} v_i}{N_I} \quad \text{(eq. 14)}$$

u_{gI} , v_{gI} : abscissa and ordinate of Ith spot center.
u_i and v_i: coordinates of pixels constructing the spot.
N_I : Number of pixels of Ith spot (area in pixels).

The goal of our work in this section is to compute the spots centers taking into account precision demand of our implementation, where the sub-pixel precision is important because its influence to distance prediction stage (see equation 2). The hardest step in center computation part is the division operations A/B in equations (13) and (14). Several methods are established to solve this problem.

A. Implementation of a hardware divider

Hardware dividers are computationally expensive and consume a considerable amount of resources, and not completely acceptable for high accuracy embedded systems. Even the completely parameterized designs like Xilinx pipelined divider [8] are not less expensive in slices (see table 2). Since, some other techniques are used to compute the center of laser spots avoiding the use of hardware dividers. A comparison between different approaches is present in section VI.

TABLE 2 : 32 BIT, FIXED POINT, XILINX PIPELINED DIVIDER IP CORE SPECIFICATIONS

Property	Divisor and dividend width	
	8	32
Number of slices	129	1666
F_{max} , MHz	385	203

B. Approximation method

Some studies suggest approximation methods to avoid implementation of hardware dividers .Such methods like that implemented in [7] replace the active pixels by the smallest rectangle containing this region, and then replace the usual division by simple shifting (division by 2).

$$u^*_{gI} = \frac{\underset{i \in I}{Max}(u_i) + \underset{i \in I}{\min}(u_i)}{2} \quad \text{(eq. 15)}$$

$$v^*_{gI} = \frac{\underset{i \in I}{Max}(v_i) + \underset{i \in I}{\min}(v_i)}{2} \quad \text{(eq. 16)}$$

This approach is approximated in equations (15) and (16), where (u_i, v_i) are the active pixel coordinates, (u^*_{gI}, v^*_{gI}) are the approximated coordinates of the spot center.

The determination of rectangle limits needs two times scanning of the image, detecting in every scanning step, respectively, the minimum and maximum of pixels coordinates.

For each spot, we should compare the coordinates of every pixel by last registered minimum and maximum to assign new values to u_m, u_M, v_m, and v_M. (m: Minimum; M: maximum). While N_p is the average area of spots (number of pixels), we can estimate the number of operations needed to calculate the center of each spot by $4N_p+6$. And in global, $N_{op} \approx 25 \times N \times (4N_P+6)$ operations are needed to calculate the centers of N spots (video-cadence 25 fps).

Such approximation is simple and easy to use but still needs considerable time to be calculated. Beside, the error is not negligible.

The error in such method is nearly 0.22 pixel, and the maximum error is more than 0.5 pixel [7]. Taking the spot of Figure 8 as an example of inaccuracy of such a method, the real center position of these pixels is *(4.47 ; 6.51)*. But when applying this approximation method, the center position will be *(5;6)*. This inaccuracy will result mismatching problem that affects the measurement result when reconstructing object.
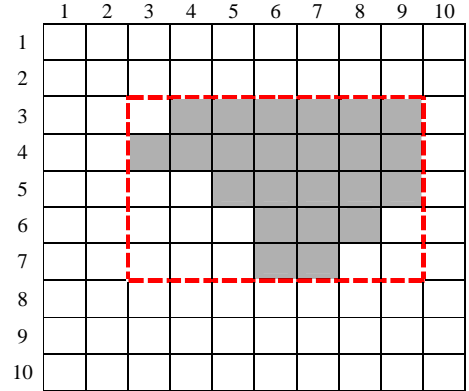

Fig. 8. Smallest rectangle containing active pixels

C. Our method

The area of each spot (number of pixels) is always a positive integer, while its value is limited in a pre-determinate interval [N_{min} , N_{max}]. Where N_{min} and N_{max} are respectively the minimum and maximum areas of laser spots in the image.

The spot areas depend on object illumination, distance between object and camera, and the angle of view of the scene.

Our method consists on realizing a FIR filter to replace the division by multiplication, to calculate the centers of each spot.

$$u_{gI}=a_1.u_1 + a_2.u_2 + \dots + a_{NI}.u_I$$
$$v_{gI}=a_1.v_1 + a_2.v_2 + \dots + a_{NI}.v_I$$

In our case, the filter coefficients a_i are constants and equals. For the spot I, with area equal N_I, the filter coefficients are: $a_1 = a_2 = \dots = a_{NI} = 1/N_I$.

In other words, it is sufficient to perform a simple convolution between inputs that are the pixels coordinates of each spot, and a short pre-determinate sequence formed of constant filter coefficients registered in a Look Up Table (see Table 3) with inverse proportionality relationship with the area of each spot, the contents of LUT are indexed by spot size.

TABLE 3: FILTER COEFFICIENTS INDEXED BY SPOT SIZE

1	2	3	4		N_{max-1}	N_{max}
1	0.5	0.25	0.125	. . .	$1/(N_{max-1})$	$1/N_{max}$

The implementation of such a filter is very easy, regarding that the most of DSP functions are provided for earlier FPGAs. For example Virtex-II architecture [10] provides an 18x18 bits Multiplier with a latency of about 4.87ns at 205MHz, and optimized for high-speed operations. Additionally, the power consumption is lower compared to a slice implementation of an 18-bit by 18-bit multiplier [10].

For N luminous spots source, the Number of operations needed to compute the centers coordinates is $N_{op} \approx 25 \times N \times N_P$, N_p is the average area of spots. When implementing our approach to Virtex II Pro FPGA (XC2VP30), it was clear that we gain in execution time and size. Comparison of different implementation approaches is described in the next section.

VI. EXPERIMENTAL RESULTS

The implementation results of distortion correction method to Xilinx Virtex II Pro FPGA (xc2vp30) are summarized in Table 4. Figure 7(a,b) presents image example before and after correction of lens distortion. Regarding size and Latency, it is clear that their results are suitable for our application, and the cost-benefits assessment is acceptable regarding high precision demands of our applications.

Comparing our method used to compute the spots centers with two other methods (see table 5), it is clear that our approach has higher accuracy, smaller size than approximation method. Since it has nearly the same accuracy of method using hardware divider, but still have a considerably small size and uses less resources. Regarding latency, the results of all three approaches respect real time constraint of video cadence (25 fps).

Comparing many measures on the depth estimation before and after the implementation of our improvements, the results indicate that the precision of the system increased, so that the residual error is reduced about 33% (see Figure 7(c)).

TABLE 4: DISTORTION CORRECTION PERFORMANCE CARACTERISTICS

Slices	1795 (13 %)
Latency	11.43 μs
Error	< 0.01 pixels

TABLE 5: CENTERS COMPUTATION PERFORMANCE CARACTERISTICS

Method	Slices	Latency (μs)	Error (pixels)
Approximation	287	4.7	0.21
Hardware divider	1804	1.77	0.0078
Our approach	272	2.34	0.015

(a) (b)

(c)

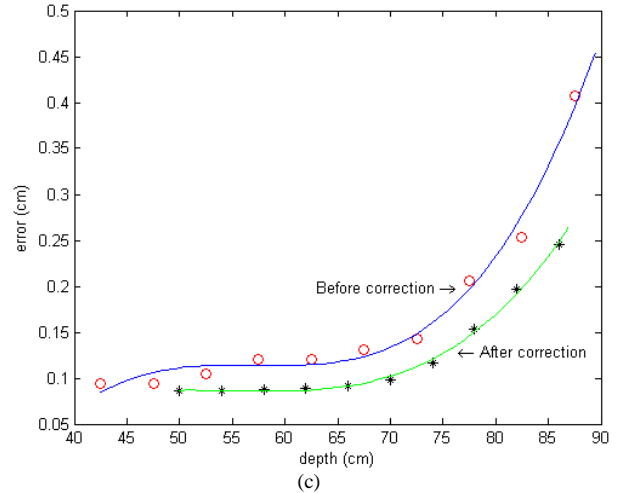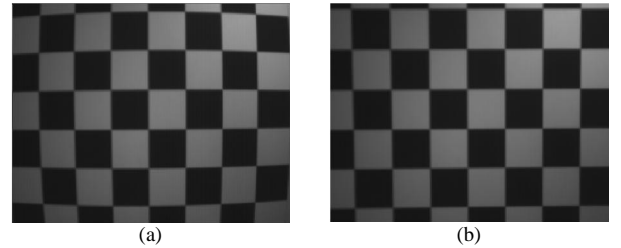Fig. 7. a)Checkerboard image before distortion correction b) Checkerboard image after correction. c) error comparison before and after applying distortion correction and centers re-computing

VII. CONCLUSION AND FURTHER WORK

In this paper, we have described an easy and simple method to compute spots centers in an active stereovision system, to guarantee accurate results respecting the constraints of an embedded system. Whereas the accuracy increased, because the error is approximately negligible. On the other hand, the correction of lens distortion will increase the response time because the model becomes fifth order polynomials. But this cost appears acceptable regarding the benefits in precision needed to reconstruct the 3D shape of

the object, where experimental results show that we reduced the residual error.

Since, we have a reliable model witch allows us to determine, with high accuracy, the 3D coordinates of the studied object from its 2D image. Our future researches will focus on recognition of this object basing on features extraction and statistical analysis of the shape,   with implementation of such theories on FPGA.

REFERENCES

[1]. T. Graba, B. Granado, O. Romain, T. Ea, A. Pinna, and P. Garda. *"Cyclope: an integrated real-time 3d image sensor"* In XIX Conference on design of circuits and integrated systems, 2004.

[2]. F. S. Marzani, Y. Voisin, L. Y. Voon and A. Diou, *"Calibration of a three-dimensional reconstruction system using a structured light source"* Optical Engineering, Vol. 41 No. 2, February 2002.

[3]. S. Woo, A. Dipanda, F. Marzani, and Y. Voisin, *"Determination of an optimal configuration for a direct correspondence in an active stereovision system"* Proc. On 2nd IASTED INT. Conf. VIIP, Malaga, Espagne, September 2002, p. 596-601.

[4]. T. Pavlidis, *"Structural Pattern Recognition"*, Springer-Verlag, Berlin (1977).

[5]. J. P. Helferty ,C. Zhang, G. Mclennan, and W. E. Higgins, *"Videoendoscopic Distortion Correction and Its Application to Virtual Guidance of Endoscopy"*, IEEE Transaction on medical imaging, vol. 20, NO. 7, July 2001.

[6]. Stuart F. Oberman, and Michael J. Flynn, *"Division Algorithms and Implementations"*. IEEE Transactions on computers, Vol. 46, No. 8, August 1997.

[7]. T. Graba, *"Etude d'une architecture de traitement pour un capteur intégré de vision 3D"*, Thèse de doctorat, Université Pierre et Marie Curie, 2006.

[8]. *"Xilinx pipelined divider V1.0"*, DS530 January 18, 2006.

[9]. W. Li, F. Boochs , F. Marzani, and Y. Voisin, *"Iterative 3D surface reconstruction with adaptive pattern projection"*, in Proc. of the Sixth IASTED International Conference on Visulatization, Imaging and Image Processing (VIIP), Palma De Mallorca, Espagne, pp.336-341, August 2006.

[10]. M. Adhiwiyogo, *"Optimal Pipelining of I/O Ports of the Virtex-II Multiplier"*, XAPP636 (v1.4) June 24, 2004.

[11]. J. Stooke, *"Cartographiy of asteroides and comet nuclei from low resolution data"* Asteroids, Comets, Meteors, pp. 583–586, 1991.

[12]. A. Kolar, T. Graba, A. Pinna, O. Romain, and B. Granado, *"Smart Bi-Spectral Image Sensor for 3D Vision"*, IEEE sensors 2007, pp295.

[13]. J. Heikkilä and O. Silven, *"A four step camera calibration procedure with implicit image correction"*, Proc. IEEE Conference on Computer Vision and Pattern Recognition, June 17-19, San Juan, Puerto Rico, 1:1106-1112.

[14]. J. Weng, P. Cohen, and M. Herniou , *"Camera calibration with distortion models and accuracy evaluation"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 14 , Issue 10 (October 1992).

[15]. J. Wang, F. Shi, J. Zhang and Y. Liu, *" A new calibration model of camera lens distortion"*, Pattern recognition (2007).

[16]. C. Slama, *"Manual of photogrammetry"*, American society of photogrammetry, Falls chuch VA, USA, 4th edition 1980.

[17]. W. Faig, *"Calibration of close-range photogrammetric systems: Mathematical formulation"*, Photogrammetric Engineering Remote Sensing, vol. 41, no. 12, pp. 1479-1486, Dec 1975.

[18]. Heikkilä, J. (1997) *"Accurate camera calibration and feature based 3-D reconstruction from monocular image sequences"* Ph.D. dissertation, University of Oulu, Finland.

[19]. "*Optimal Pipelining of I/O Ports of the Virtex-II Multiplier*" XAPP636(v 1.4), June 24, 2004.

[20]. Jean-Yves Bouguet, *"Camera Calibration Toolbox for Matlab"*, http://www.vision.caltech.edu/bouguetj/calib_doc/