Geometrical Analysis of Localization Error in Stereo Vision Systems

Fahimeh Fooladgar, Shadrokh Samavi*, S. M. Reza Soroushmehr and Shahram Shirani

Abstract—Determining an object location in a specific region is an important task in many machine vision applications. Different parameters affect the accuracy of the localization process. The quantization process in CCD of a camera is one of the sources of error which causes estimation rather than identifying the exact position of the observed object. A cluster of points, in the field of view of a camera are mapped into a pixel. These points form an uncertainty region. In this paper, we present a geometrical model to analyze the volume of this uncertainty region as a criterion for object localization error. The proposed approach models the field of view of each pixel as an oblique cone. The uncertainty region is formed via the intersection of two cones, each emanating from one of the two cameras. Because of the complexity in modeling of two oblique cones’ intersection, we propose three methods to simplify the problem. In the first two methods only four lines are utilized. Each line goes through the camera’s lens, modeled as a pinhole, and then passes one of the four vertices of a square that is fitted around the circular pixel. The first proposed method projects all points of these four lines into an image plane. In the second method, the line-cone intersection is utilized instead of intersection of two cones. Therefore by applying line-cone intersection, the boundary points of the intersection of the two cones are determined. In the third approach, the extremum points of the intersection of two cones are determined by the Lagrangain method. The validity of our methods is verified through extensive simulations. Also, we analyzed effects of parameters, such as the baseline length, focal length, and pixel size, on the amount of the estimation error.

Index Terms—Cone intersection, geometrical error analysis, quantization error, stereo vision.

I. INTRODUCTION

Obtaining an accurate measurement of a 3D position via stereo vision is one of the most important issues in the field of computer vision [1]. There are three main stages in stereo vision: determining external and internal camera parameters (calibration), specifying corresponding pixels of two images and finally calculating 3D position of an object by triangulation [2]. One of the fundamental problems in this field is triangulation which refers to reconstruction of a 3D position from its 2D projected points on two or more cameras. Two corresponding pixels in two cameras are the projections of an object point in the 3D space. The 3D position of that point can be calculated by intersecting two corresponding lines of sight from the two cameras where each line passes through the optical center of a camera and the center of the pixel in that camera. But in the presence of noise, these lines of sight may not intersect in the 3D space. Therefore several methods have been developed to rectify this problem. The methods investigated in [3]-[9] are some examples of these solutions. These methods are developed to displace the corresponding pixel in a manner that the two lines of sight intersect with each other. In [7], as shown in Fig.1 (a), the midpoint of a line which is perpendicular to these two lines is chosen as the intersection point of these rays. Due to various approximations, this method does not generate an optimal result. Therefore, optimal methods have been proposed in [5]-[7], [9] to improve this method. In optimal methods, the displacements of corresponding pixels are minimized, as shown in Fig.1 (b).

To acquire better position estimation, errors in all three steps of stereo vision should be reduced because small and negligible errors in the near field result in large errors in the far field. Recent works have been investigating the pixel correspondence problem. Therefore, significant precision to determine matching pixels between images is achieved [10]. Another source of localization error is due to pixel quantization and limitation of spatial resolution in CCD of cameras. Authors of [1] proposed a method to estimate localization errors via intersection of projected pyramids originating from every two corresponding pixels of the two images. They utilized the volume of the intersection region as the estimation error.

Since in traditional stereo vision the baseline length is fixed, the error is increased with the growing distance of an object from cameras. In [11] a variable baseline stereo system is presented which consists of high speed vision and high speed linear slider. In this system by moving slider horizontally, the baseline length can be varied. Also, a method to control the baseline length based on its distance to the object is proposed. The method of multi-baseline and multi-resolution stereo is addressed in [10]. In this approach baseline and resolution of
stereo cameras can be varied proportional to the depth such that the depth error remains constant.

In [12], parameters of a stereo system are evaluated to minimize the 3D reconstruction error. The authors of [12] derived the mathematical relationship between the variance of error and the parameters of a stereo system. They classified the error model of a stereo system in two categories. In the first one, the quantization error or the worst case analysis are considered such that the lower and upper bounds of the error are computed. In the second category, the Gaussian error model is investigated to analyze the stereo error. Wenhardt et al. in [13] presented a 2D model and also a 3D model to develop mathematical relationships between the error and the parameters of a stereo system. In the 2D model, the optimum baseline and the focal length are calculated by minimizing the error and for the 3D model by utilizing the Monte Carlo simulation.

Malik and Bajcsy in [14] have looked at the placement problem of a 3D stereo camera in a monitoring area. They located cameras in an appropriate position not only to decrease the measurement error but also to improve the resolution of captured images. Although two cameras which are close to each other and also close to an object have suitable resolutions, they produce a large depth estimation error. In contrast, if two cameras are very far from each other, the depth estimation error is reduced but proper resolutions cannot be obtained. Therefore, in order to attain good resolution and low error, authors of [14] utilized genetic algorithm and gradient descent algorithm to place cameras in appropriate positions.

In some applications of visual sensor network (VSN) random distribution of cameras are utilized. This is done in applications such as environmental monitoring and surveillance of remote and inaccessible areas where optimum placement of cameras is not possible. In such a network, cameras have different positions, orientations and also different levels of energy. Therefore one needs to select some of them for target localization. The authors of [15] and [16] suggested an approach for target localization in VSNs. Their approach is based on the tradeoff between the accuracy of target localization and the energy consumption in VSNs. By using Shannon entropy definition to quantize the reduction in the entropy of target position, an entropy based utility function is constructed which is used as a measure of accuracy of target localization. In these two papers the priori probability distribution of target location is considered to be known.

Minimization of localization error is investigated by Isler and Bajcsy in [17]. They consider a 2D generic sensor model where measurements can be considered as polygonal, convex subsets of the plane. The ultimate goal of [17] is to select the minimum number of cameras such that the quality of localization is improved. The authors of [18] address the target localization problem in VSNs. In their method, pieces of information from a set of cameras’ field of views (FoV) are fused, and then the non-occupied areas of their FoVs are investigated as the possible locations of targets. This approach is efficient to localize crowded target environments. Moreover, error analysis for stereo vision could be considered in three aspects of geometrical, statistical, and visual qualities [19].

In this paper we consider a geometrical analysis and propose a model to analyze the quantization error in CCD of cameras. We then determine the localization error due to the quantization limitation. In our proposed model, we consider each pixel as a circle and use the projected cone to represent the FoV of each pixel. Thus to determine the amount of error in the object localization via two cameras, the intersection of the corresponding cones from the two cameras is utilized. Since finding the intersection of two cones, and especially two oblique cones, is very complex, we propose three methods to simplify this process. In the first proposed method, the projection of all points of a line on a circular pixel is considered. In the second method, instead of considering the intersection of two cones, the intersection of a line and a cone is utilized. In the third proposed method, the Lagrangian method is employed and the maximum and minimum points of the intersection region are found along each of the three directions. Since the FoV of each pixel is considered as a cone, the proposed model can easily be applied to any configuration of cameras. Another contribution of this paper is that the proposed methods can be generalized to a multi-camera system, where more than two cameras are used to have more accurate estimation of an object’s location. In such a system, each added camera leads to a cone intersection formation and the results of these intersections determine the

---

**Fig.1. Triangulation: a) midpoint method, b) optimal method [4].**
boundary points of the intersection region. Consequently, the goal of our paper is to develop general and scalable methods to calculate localization error. By generality, our model can be used not only for the conventional stereo setups, but also for cameras with any position and orientation. Furthermore, our methods are scalable such that it could be applied for multi-camera systems.

The rest of this paper is organized as follows. In section II, the problem statement is presented. There, the camera model, perspective projection model and also the localization error are studied. In section III, the mathematical models are represented which are the bases of the methods proposed in section IV. The simulation results of the proposed model are evaluated in section V and concluding remarks are given in section VI.

II. PRELIMINARY PRINCIPLES

In this section, the camera model, perspective projection and the object localization are described.

A. Camera Model

In this paper pinhole camera model is used. Fig. 2 depicts FoV of a pinhole camera model in a camera coordinate system (CCS). As shown in Fig. 2, FoV of a camera is defined as a pyramid or a cone. The formation of cone or pyramid depends on the assumption of circular or rectangular pixel. The coordinates of the cone or pyramid's vertex are equal to the coordinates of the camera's focal point which are assumed to be the center of the CCS. Coordinates \((X_c, Y_c, Z_c)\) in Fig. 2 denotes the position of a camera (i.e. \(i^{th}\) camera in a VSN) in the world coordinates system (WCS). A camera is represented by a six-tuple \((X_i, Y_i, Z_i, \theta_i, \varphi_i, \phi_i)\) in which \((X_i, Y_i, Z_i)\) defines the camera's position on the monitoring area and \(\theta_i, \varphi_i, \phi_i\) are the rotation angles of the camera around the \(X_w, Y_w\) and \(Z_w\) axes. These six parameters are the camera parameters and are obtained via a camera calibration process.

In this paper, subscripts \(w\) and \(c\) respectively belong to variables in the world and camera coordinate systems. Also, capital letters indicate variables in the world and camera coordinate systems while lowercase variables are reserved for the image plane coordinate system.

B. Perspective Projection Model

Human visual system and also every machine vision system, such as a camera, perform transformation from 3D real space to a 2D space. This transformation is done by perspective projection [20]. In the perspective projection model, a pinhole camera is supposed to map an arbitrary point \(P_w = [X_w, Y_w, Z_w]^T\) from the 3D-WCS to a point \((u, v)\) on its image plane. By applying the rotation and translation matrix of the camera, the point \(P_w\) in WCS can be translated to CCS as \(P_c = [X_c, Y_c, Z_c]^T\) via (1).

\[
\begin{align*}
(X_c, Y_c, Z_c) &= R \left(\begin{array}{c} X_w - X_o \\ Y_w - Y_o \\ Z_w - Z_o \end{array}\right) \\
(1)
\end{align*}
\]

where \(O = [X_o, Y_o, Z_o]^T\) is the center of CCS in WCS and is used for translation purpose while \(R = R_\theta R_\varphi R_\phi\) is the rotation matrix [20]. This matrix is the combination of three rotation angles (ie. \(\theta, \varphi, \phi\) ) which are the rotation angles of the camera coordinates axes around the world coordinates axes. Therefore the point \((u, v)\) on a camera image plane is calculated as follow:

\[
u = \frac{X_c}{Z_c}, \quad v = \frac{Y_c}{Z_c}
\]

(2)

where \(f\) is the focal length of the camera.

C. Object Localization

Suppose that we have an object point at position \(P_w\) in the WCS. As illustrated in Fig. 3, this object is mapped on the image plane of the camera \(O_1\), at point \(p_1 = (u_1, v_1)\). Subscripts \(l\) and \(r\) respectively denote the left and right

---

**Fig. 2:** Camera Field of View (FoV).

**Fig. 3:** Intersection of two lines in a stereo system.
cameras. In other words, in Fig. 3, line \( L_1 \), joining the focal point of the camera \( O_l \) and the object point, intersects the image plane of the left camera at \( p_l \). All points of the line \( L_1 \) are projected into pixel \( p_l \). Hence by using only one camera, the object position cannot be estimated. In order to estimate coordinates of the object, \( P_w \), at least two cameras are necessary.

Suppose that the goal is to estimate the target position, \( P_w \), in a stereo vision system where two cameras, left and right, are available and also the projections of the target point on two cameras are known. In such a case, as shown in Fig. 3, the point \( P_w \) is mapped into points \( p_l = (u_l, v_l) \) and \( p_l = (u_l, v_l) \) on the right and left cameras respectively. In an ideal situation, each point in the 3D-WCS corresponds to a specific point on the image plane and also there is no noise. Then the target point can be exactly determined by intersecting two lines, each line joins the focal point of one camera and the projected point on its image plane. Fig. 3 depicts the intersection of these lines. But in a real camera system, there is no one-to-one correspondence between the points in a WCS and the ones on a camera image plane. Hence, an object point cannot be determined uniquely. For example, due to pixel quantization and limitations of spatial resolution in the image plane, the two mentioned lines do not necessarily intersect or they might intersect in a point other than \( P_w \). Therefore, instead of identifying just one target point, a 3D uncertainty region, is formed where each point in that region can be considered as the object point.

In [1], the quantization error is modeled as the projected pyramids and the uncertainty region is the intersection of the pyramids. In [1], each point on the boundary of uncertainty region is obtained via the intersection of one of the four side edges of one pyramid with one of the four sides of the other pyramid. Therefore, 16 intersection operations per pyramid and a total of 32 intersections for two pyramids must be performed. Since all of the intersection points are not useful, a method is used to search for the true vertex points. Then the set of intersection points is obtained and in order to compute the volume of the region for error estimation, an ellipsoid is fitted on the vertices of the region. The ellipsoid fitting is performed using principal component analysis (PCA) or the set of points could form a convex hull whose volume can be computed using the method mentioned in [21]. Hence, the higher volume of this region indicates a higher uncertainty in estimating the target point. A two dimensional view of the uncertainty region produced by intersection of FoV of two cameras is illustrated in Fig.4.

III. MATHEMATICAL MODELS

To calculate the localization error caused by pixel quantization, the boundary points of the uncertainty region must be found. Therefore, we propose a geometrical model to analyze this error. We consider circular pixels, which makes FoV of each pixel to be modeled as an oblique cone. Therefore, rather than considering equations of the faces of a pyramid, we utilize spatial equation of a cone. Fig.5 illustrates a conical volume corresponding to each pixel. In this figure it is assumed that the base of the cone is a circular pixel, onto which the points of the uncertainty region are projected.

In our model, we use the equation of an oblique cone in 3D WCS. Therefore, in this section, the underlying mathematical models are presented. To come up with the equation of an oblique cone in 3D WCS, let us first define a circle and then define a line. General formula of a circle is defined as:

\[
(x - x_0)^2 + (y - y_0)^2 = r^2 \tag{3}
\]

where \((x_0, y_0)\) is the center of a circle and \(r\) is its radius. The equation of a line, passing through two points, can be defined as:

\[
\frac{X - X_1}{n_1} = \frac{Y - Y_1}{n_2} = \frac{Z - Z_1}{n_3} \tag{4}
\]

In this equation, \(N = (n_1, n_2, n_3)\) is the direction vector of the line which is calculated as \(N = P_2 - P_1\) where this line goes through points \(P_1 = (X_1, Y_1, Z_1)\) and \(P_2 = (X_2, Y_2, Z_2)\). The parametric form of a line equation is defined by (5):

\[
\begin{align*}
X &= n_1 t + X_1 \\
Y &= n_2 t + Y_1 \\
Z &= n_3 t + Z_1
\end{align*} \tag{5}
\]
where \( t \) is the line parameter. The line can be written in a matrix form which is defined as:

\[
P = Nt + P_c
\]  

(6)

Let us assume that we have a circle in the plane \( Z = Z' \) with radius \( r \) where its origin is at point \((X',Y')\). The equation of this circle is defined as:

\[
\begin{align*}
(X - X')^2 + (Y - Y')^2 &= r^2 \\
Z &= Z'
\end{align*}
\]  

(7)

Using (7), each pixel in the camera image plane can be defined as a circle with a different origin. Hence, a bundle of straight lines passing through the optical center of the camera and the circular pixel define an oblique cone (Fig. 5) which is described by the following equation [22]:

\[
P_c^T Q_c P_c = 0
\]  

(8)

where \( P_c = [X_c, Y_c, Z_c]^T \) and \( Q_c \) is defined in CCS by:

\[
Q_c = \begin{pmatrix}
1 & 0 & -\frac{X'}{Z'} \\
0 & 1 & -\frac{Y'}{Z'} \\
-\frac{X'}{Z'} & -\frac{Y'}{Z'} & \frac{X'^2 + Y'^2 - r^2}{Z'^2}
\end{pmatrix}
\]  

(9)

In this equation, it is assumed that the Z-axis is the optical axis of the camera. Also, Equation (8) can be defined in a WCS as follows:

\[
P^T Q_o P = 0
\]  

(10)

In Equation (10), \( P = P_w - O_r \) where \( O_r \) is the vertex of the cone which is the optical center of the camera in the WCS. Also \( Q_o \) is defined by:

\[
Q_w = R^T Q_o R
\]  

(11)

where \( R \) is a rotation matrix of the camera. The base of this cone is considered to be in the image plane as a circle with radius \( r \). Also it is assumed that \((X',Y',Z')\) is the center of the circle and \( Q_c \) is defined by (9).

IV. PROPOSED METHODS

To form the uncertainty region and utilize its volume as a metric for the localization error, two cones corresponding to the two pixels, should be intersected. However, computing the volume of the intersection region is complex. Hence, in this section we present three approaches to calculate the boundary points of the uncertainty region based on the described mathematical models. All three approaches are solutions for one geometrical problem and hence produce same results. The only differentiating aspect would be the computational complexity of each individual approach. Also, it is assumed that we know which two pixels, in the two cameras, receive the image of an object point. Consequently, the aim of the methods of this section is to decrease the computational complexity of the localization error estimation through some approximations.

A. Line-Circle Projection Method

Suppose that we have two corresponding pixels, represented by two circles in the left and right cameras. In this method, instead of intersecting two cones, we perform the following two steps. In the first step, we consider the pixel in the right camera and also four lines coming from the pixel in the left camera. Each of these lines passes through the optical center of the left camera and one of the vertices of a square, fitted outside the corresponding circular pixel. Then we project these four lines onto the image plane of the right camera. After that, we investigate which points of these lines are projected onto the corresponding circular pixel in the right camera. In the second step, we switch the roles of the two pixels and the lines would come from the right camera and are seen by the pixel in the left camera. Consequently, the boundary points of the uncertainty region are computed.

To describe this method mathematically, we consider an arbitrary line \( L_1 \) in the left camera and the corresponding pixel \( p_r \) in the right camera (Fig. 6). Line \( L_1 \) passes through the optical center of the left camera \((O_l)\) and one of the vertices of the square fitted outside pixel \( p_l \). The line equation for \( L_1 \) is defined as:

\[
\begin{align*}
X_l &= n_1 t + O_{l1} \\
Y_l &= n_2 t + O_{l2} \\
Z_l &= n_3 t + O_{l3}
\end{align*}
\]  

(12)

where \( O_l = (O_{l1}, O_{l2}, O_{l3}) \) is an optical center of the left camera and \( N = (n_1, n_2, n_3) \) is the direction vector of \( L_1 \) defined as:

\[
N = P_a - O_l
\]  

(13)

where \( P_a \) is one of the vertices of the square-shaped pixel. Then all points of line \( L_1 \) are translated into the right CCS as follow:

\[
\begin{align*}
X_c &= r_{11} t_1 + r_{12} t_2 + r_{13} t_3 + n_1 t + O_{l1} - O_{1r} \\
Y_c &= r_{21} t_1 + r_{22} t_2 + r_{23} t_3 + n_2 t + O_{l2} - O_{2r} \\
Z_c &= r_{31} t_1 + r_{32} t_2 + r_{33} t_3 + n_3 t + O_{l3} - O_{3r}
\end{align*}
\]  

(14)

where each \( r_{ij} \in R \) (for \( 1 \leq i, j \leq 3 \)) is an element of the rotation matrix of the right camera and \( O_c \) is the optical center of the right camera. Then all points of line \( L_1 \) which are translated into the right CCS must be projected on the right camera image plane as follow:

\[
\begin{align*}
u_r &= f \frac{X_c}{Z_c} = f r_1 (n_1 t + O_{l1} - O_{1r}) \\
v_r &= f \frac{Y_c}{Z_c} = f r_2 (n_2 t + O_{l2} - O_{2r})
\end{align*}
\]  

(15)

In (15), \( f \) is the focal length of the right camera.

The inequality of (16) identifies the points of line \( L_1 \), which are projected into pixel \( p_r \) in the right camera.

\[
(u_r - i_p)^2 + (v_r - j_p)^2 \leq r^2
\]  

(16)

This inequality determines a circular pixel in the image plane coordinates system of the right camera. Also, coordinates \((i_p, j_p)\) define the center of the pixel \( p_r \) and \( r \)
is the radius of pixel. By substituting (15) into (16), the only unknown variable of this inequality is \( t \) (line parameter). Using (16), the boundary points of the part of the line which are projected on the circular pixel \( p_r \) are calculated.

The above mentioned steps must be done for all four lines from the left camera and the four from the right camera. Consequently, the boundary points of the uncertainty region are obtained.

B. Line-Cone Intersection Method

In this method, we also consider each pixel as a circle but use intersection of a cone and a line instead of the line-circle projection. Therefore, the spatial equation of a cone is utilized where each circular pixel is the cross section of a cone. To estimate an object’s position via two cameras, we have two corresponding pixels and the resulting two cones. To describe the proposed line-cone intersection, we can intersect the cone from the right camera by four lines from the left camera. Each of these four lines passes through the optical center of the left camera and one of the four vertices of a square, fitted outside the circular pixel. Then this process is repeated for the corresponding cone in the left camera and four lines from the right camera.

This method is equivalent to the line-circle projection method. This is because all points in the 3D space which are projected into a specific 2D circle (pixel) generate a 3D cone. The vertex of this 3D cone is at the origin of CCS and its base is a circular pixel in the camera image plane. Hence this method is the simplified model of the line-circle projection method.

To describe the line-cone intersection, we consider a cone in the right camera, as defined in (10), and four lines in the left camera. Each line coming from the left camera is defined as

\[
P_w = Nt + O_l
\]

where \( N \) is a direction vector of the line and \( O_l \) is an optical center of the left camera. To find the intersection points of the line and the cone, we first substitute \( P_w \) in the quadratic Equation (10). Then we simplify the result to obtain

\[
c_2^2 t^2 + 2c_1 t + c_0 = 0, \quad \text{where} \quad \Delta = O_l - O_r, \quad c_2 = N^T Q_w N, \quad c_1 = N^T Q_w \Delta \quad \text{and} \quad c_0 = \Delta^T Q_w \Delta.
\]

By solving this equation, two intersection points are determined. Since some of these points may be located in the reflecting cone, the condition defined in (18) must be satisfied for all of these points to eliminate the points on the reflected cone.

\[
P_w - O_r \geq 0
\]

C. Lagrangian Method

In this section, to find the boundary points of the uncertainty region we propose a method to calculate the maximum and minimum points on the cones’ intersection region in all three directions. The Lagrangian method provides a strategy to find maxima and minima (i.e. extrema) of a function subject to one or more constraints. Suppose we have an optimization problem which its goal is to find the extreme points of function \( z = f(x,y) \) subject to \( g(x,y) = 0 \). In such a problem, the Lagrangian method can be utilized. In the following, the extremum points of the intersection of two cones are calculated by this method. A two dimensional example of extremum points of a two-cone intersection is illustrated in Fig.4.

Suppose that \( p_l \) and \( p_r \) are two corresponding pixels in the left and right cameras respectively, and their cones are represented by \( F_l(X,Y,Z) = 0 \) and \( F_l(X,Y,Z) = 0 \). These equations can be extracted from (10) and are expressed in (19) and (20) respectively.

\[
F_l(X,Y,Z): \quad q_{11}X^2 + q_{22}Y^2 + q_{33}Z^2 + 2q_{12}XY + 2q_{13}XZ + 2q_{23}YZ = 0
\]

\[
F_l(X,Y,Z): \quad q_{11}X^2 + q_{22}Y^2 + q_{33}Z^2 + 2q_{12}XY + 2q_{13}XZ + 2q_{23}YZ = 0
\]

In these equations, \( q_{ij} \in Q_w \) for \( 1 \leq i,j \leq 3 \).

To find the boundary of the uncertainty region, the cones corresponding to these two pixels must be intersected. By separation of \( Z \) in (19) from \( X \) and \( Y \), this equation could be represented as \( Z = G_l(X,Y) \) which is defined as

\[
G_l(X,Y) = \frac{1}{q_{33}} \times (-q_{13}X - q_{23}Y \pm \sqrt{(q_{13}X + q_{23}Y)^2 - q_{33}(q_{11}X^2 + q_{22}Y^2 + 2q_{12}XY)})
\]

Substituting \( Z \) defined by (21) into (20) results in (22). Equation (22) is a function of \( X \) and \( Y \), which we call it as \( G_l(X,Y) \).

\[
G_l(X,Y) = F_l(X,Y,G_l(X,Y)) = 0
\]

Hence, for acquiring the intersection points, the extremum points of \( G_l(X,Y) \) are determined such that the points would satisfy (22). Therefore the Lagrangian method can be utilized. In this method the Lagrangian function is defined by

\[
L(X,Y) = G_l(X,Y) + \lambda G_l(X,Y)
\]

where \( \lambda \) is known as the Lagrange multiplier. According to this method, the system of three equations with three variables, as shown in (24), must be solved.
The boundaries of the uncertainty region are found using the three proposed methods. In the first method, projection of a line into a circle is represented. In this method, the problem is modeled by a 3D line and a 2D circle. In the second method, line-cone intersection method is used instead of the two-cone intersection. In this method the equation of a line and an oblique cone in the 3D space are used. In essence, these two methods are equivalent. The only difference between these two methods is their mathematical models. In the third method, the extremum points of intersection of two cones are calculated by the Lagrangian method. For this purpose, the cone equation in the 3D space is utilized. In terms of the complexity of the model, the Lagrangian method has simpler model than the other two proposed methods.

When intersecting two cones, the Lagrangian method is the general case of the three proposed methods. Therefore, this method can cover both the Line-Cone Intersection and the Line-Circle Projection methods. However, the drawback is its required computational intensity which is more complex than the other two proposed approaches.

This method tries to solve the system of three equations with three unknown variables. This cannot be solved analytically and numerical solutions are required. Features of the three proposed methods are summarized in TABLE I.

Using the rotation and translation matrices of each camera, models of all three methods are written in WCS. Therefore, all approaches can be utilized for any camera with any configuration; demonstrating the generality of our model. Also, all methods can be used for multi camera system, in which by adding a camera, the corresponding pixel of the object in its image plane is added. In other word, suppose we want to estimate the uncertainty bound caused by three cameras. To find the intersection points of the cones, the previously described Lagrangian method must be applied for each pair of cones. Then the bounding points of the intersection will be obtained.

### E. Comparison of the computational Complexities

In this section, the computational complexities of the pyramids intersection approach [1] and the three proposed methods are investigated. All comparisons are based on the number of basic arithmetic operations performed to solve the intended equations. We consider that the addition-subtraction and multiplication operations between two n-bit numbers respectively have complexities of $A(n)$ and $M(n)$ where $A(n) < M(n)$. Also, division and square root operations have the same complexities as the multiplication operation [23].

For the pyramids intersection method, each intersection point is obtained via crossing of two adjacent surfaces of one pyramid and intersecting it with one of the faces of the other pyramid. Therefore, 16 intersection operations per pyramid, and a total of 32 intersections, must be performed. Since not all intersection points are involved in formation of the uncertainty region, in [1] a routine is proposed to find the appropriate points. In each of the 32 intersections, a system of three linear equations with three unknown variables, similar to Equation (25), must be solved.

$$
\frac{\partial L}{\partial X} = 0 \\
\frac{\partial L}{\partial Y} = 0 \\
G_i(X, Y) = 0
$$

By solving the above system of equations, the extreme points of the objective function, $G_i(X, Y)$, are determined. By substituting these points into $Z = G_i(X, Y)$, the global maximum and minimum points of the intersection region, along the $Z$ direction, can be specified. To find the extreme points of the intersection region along the other two directions, the same method is carried out and therefore a similar system of nonlinear equations should be solved. Thus the set of points are obtained which are the extreme points of the intersection region of the two cones.

### D. Advantages and Disadvantages of Proposed Methods

The characteristics of three proposed methods are listed in TABLE I.

<table>
<thead>
<tr>
<th>Method</th>
<th>Geometrical Model</th>
<th>Complexity</th>
<th>Mathematical Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line-Circle Projection</td>
<td>3D Line , 2D Circle</td>
<td>8 quadratic inequalities</td>
<td>Analytical solution</td>
</tr>
<tr>
<td>Line-Cone Intersection</td>
<td>3D Line , 3D Cone</td>
<td>8 quadratic equations</td>
<td>Analytical solution</td>
</tr>
<tr>
<td>Lagrangian Method</td>
<td>3D Cone</td>
<td>Three systems of three equations</td>
<td>Numerical solution</td>
</tr>
</tbody>
</table>

For the pyramids intersection method, each intersection point is obtained via crossing of two adjacent surfaces of one pyramid and intersecting it with one of the faces of the other pyramid. Therefore, 16 intersection operations per pyramid, and a total of 32 intersections, must be performed. Since not all intersection points are involved in formation of the uncertainty region, in [1] a routine is proposed to find the appropriate points. In each of the 32 intersections, a system of three linear equations with three unknown variables, similar to Equation (25), must be solved.

$$
\begin{pmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23} \\
a_{31} & a_{32} & a_{33}
\end{pmatrix}
\begin{pmatrix}
X \\
Y \\
Z
\end{pmatrix}
= 
\begin{pmatrix}
b_1 \\
b_2 \\
b_3
\end{pmatrix}
\rightarrow
P_w = A^{-1}b
$$

Solving of Equation (25) requires an inversion of the matrix $A$. The number of basic operations to compute the inversion of a $3 \times 3$ matrix would be $27M(n) + 14A(n)$. Therefore, to solve Equation (25) we need $27M(n) + 14A(n)$ operations for the matrix inversion and $9M(n) + 6A(n)$ operations for multiplication of $A^{-1}$ and $b$. Hence, to compute $P_w$, the required number of operations is $27M(n) + 14A(n) + 9M(n) + 6A(n) = 36M(n) + 20A(n)$. After solving (25), each of the 32 computed points must be checked against (26). The condition in (26) checks if a point is outside a plane or not. This condition must be tested for all 8 planes of two pyramids to make sure that each point is located in all of them.

$$
\text{point} = \begin{cases}
\text{vertex} & a_{i1}X + a_{i2}Y + a_{i3}Z - b_i \geq 0 \\
\text{nonvertex} & \text{otherwise}
\end{cases}
$$

Hence, the number of operations for testing each of these 32 intersection points is $3(3M(n) + 3A(n)) = 24(M(n) + A(n))$. Consequently, the total number of operations to
compute one intersection point is $60M(n) + 44A(n)$.

In the two proposed Line-Cone and Line-Circle methods, 8 quadratic equations of the form $a^2t^2 + bt + c = 0$ must be solved. In general, two intersection points are determined by solving this equation and substituting the variable $t$ into the corresponding line equation of $p_w = Nt + O$. To solve the quadratic equation via the quadratic formula, $(-b \pm \sqrt{b^2 - 4ac}/2a)$, we need $7M(n) + 3A(n)$ operations to determine the two values of $t$. Also, $6M(n) + 6A(n)$ operations are required to determine the two intersection points. Hence, the number of operations for one intersection point is $(13M(n) + 9A(n))/2$. The condition defined in (18) must be satisfied for all of these points to eliminate the points on the reflected cone. This condition requires $3A(n)$ operations. Consequently, the total number of operations to compute one intersection point is $6.5M(n) + 7.5A(n)$.

The proposed Lagrangian method requires solution of 3 systems of three quadratic equations which demands much higher complexity than the first two proposed methods. We compared the complexities of the first two proposed methods with that of the pyramid-intersection algorithm. These are summarized in Table III. We see that the proposed methods require much lower number of operations than the pyramids intersection method.

**V. SIMULATION RESULTS**

In this section, we present two sets of experiments to evaluate the performance of each of the proposed methods. To evaluate them, we implemented our methods in MATLAB and numerically solved Lagrangian equations by the Maple software. In the first set of simulations, the effects of internal and external parameters of cameras on measurement error in parallel stereo setup are investigated. These parameters are baseline length, pixel size, focal length and distance of target to cameras. Furthermore the effects of object position in this error are evaluated. In the second set of simulations, the correctness of our method is evaluated and compared with the pyramids intersection method of reference [1]. All three proposed approaches are the solutions for one mathematical problem with different advantages and disadvantages as was described in section IV.

Hence, in the simulation, the results of the three proposed methods are similar and only one graph is illustrated, representing all of the proposed methods. The parameters of cameras are set according to TABLE III. In all simulations, maximum and minimum values of points in the uncertainty region, along each of the three directions, are calculated and multiplied to construct the volume of the uncertainty region.

In Fig. 7 to Fig.10, the results of the volume of the uncertainty region as a function of the mentioned parameters are shown.
In this figure, the vertical axis shows the measurement error and the horizontal axis displays pixel size. Fig. 10 indicates the relationship between the error and the distance between the target and the cameras. While target moves away from the cameras, the quantization error increases. The closer the target to the center of the image plane, the larger the cone’s surface and hence the greater error would be generated.

In two other simulations (Fig. 11 and 12) only the line-cone intersection method is implemented. Fig. 11 illustrates the effect of the distance between two cameras on the measurement error. This simulation is also performed for four different object positions. When the length of the baseline of the two cameras is increased and the depth of object in 3D space is decreased then the estimation error is reduced. This is because, as the baseline length decreases and the depth of the object increases, the object point is mapped on a point in the image plane which is closer to the center of the image plane. Therefore, the cone gets wider and the intersection region
becomes larger, which results in larger uncertainty region. Fig. 12 illustrates the estimation error for all object points which are in the plane parallel to the camera plane. In this experiment the specifications and locations of cameras are based on TABLE III. The object points are in the plane \( z = 100 \). As the simulation results show, the object points which are mapped farther from the center of cameras have lower error. In general, the cones which pass the pixels farther from the center of image plane have smaller angle of view than the cones which pass the pixels around the center of image plane. Therefore, those object-points that are mapped farther from the centers of the image planes of the two cameras are seen by narrower corresponding cones which are emanated from the two pixels. Hence, the volume of the uncertainty region, corresponding to such object-points, would be smaller.

To validate the correctness of the proposed approaches, the relative error is calculated in the following manner:

\[
E_{\text{Relative}} = \frac{\left\| P_{w, \text{true}} - P_{w, \text{estimated}} \right\|}{\left\| P_{w, \text{true}} \right\|} \tag{27}
\]

When performing simulations the true position of the target is known. In Equation (27), \( P_{w, \text{true}} \) is the real position of the object in 3D space and \( P_{w, \text{estimated}} \) is the estimated position of the object which considered as the center point of the uncertainty region.

In this experiment, the specifications and locations of the cameras are specified in TABLE III. In these simulations, the baseline length is changed and in each step, the object position in the overlapped FoV of the two cameras is randomly changed. Then every time that the object position is changed the relative error, \( E_{\text{Relative}} \), is calculated. Hence, for each baseline length the average of relative errors is computed. Fig. 13, illustrates the relative versus the baseline length. As shown in Fig. 7, while the baseline length increases, the volume of the uncertainty region becomes smaller. Therefore, the error in estimation of the object position is also decreased. Hence, as expected, \( E_{\text{Relative}} \) is also decreased.

To compare the correctness of our proposed methods, the pyramids intersection method [1] is also utilized. The pyramids intersection method models the uncertainty region produced from the quantization error of pixels. Fig. 13 shows the relative error produced by both the cones intersection and the pyramids intersection methods. The accuracies of both methods are almost equal.

VI. CONCLUSION

In this paper we presented a geometrical approach to estimate the amount of the localization error. To calculate the mentioned error, we used a conical intersection region, where the cones are originated from the corresponding pixels of two cameras. Since computing the intersection of two cones is difficult, we proposed three methods to simplify this problem. Indeed, in this paper three approaches for finding the bounding points of two cones intersection in all three directions were proposed. We also investigated the generality and also scalability of the proposed model. Also through simulations, we showed that the error metric has direct relationship with the distance of the camera to the target and direct relationship with the pixel size. It was also shown that the error has inverse relationship with both the baseline length and the focal length. While our methods’ results were similar to the two pyramids approach, the two proposed methods of Line-Cone Intersection and the Line-Circle Projection have much lower computational burden. Also, the proposed Lagrangian method is the general case which can cover the other approaches, but it is computationally more complex.

REFERENCES


Fahimeh Fooladgar received a B.S. degree in computer engineering (magna cum laude) in 2009 and a M.S. degree in computer architecture (honors) in 2012, both from Isfahan University of Technology, Isfahan, Iran. Her research interests include mathematical modeling of visual sensor networks and computer vision.

Shadrokh Samavi (M’07) is a Professor of Computer Engineering at Isfahan University of Technology, Iran and an Adjunct Professor at the ECE department of McMaster University, Canada. He completed a B.S. degree in Industrial Technology and received a B.S. degree in Electrical Engineering at California State University, a M.S. degree in Computer Engineering at the University of Memphis and a Ph.D. degree in Electrical Engineering at Mississippi State University, U.S.A. Shadrokh Samavi’s research interests are in the areas of video and image processing as well as hardware implementation and optimization of image processing algorithms.

S. M. Reza Soroushmehr received his B.S. degree from Isfahan University of Technology (IUT), Iran in June 1999 in the field of hardware engineering. He then received his M.S. degree in computer architecture from IUT in 2003. He is currently a visiting scholar at the department of electrical and computer Engineering of McMaster University, Canada. He is also a student member of IEEE. His research interests are visual sensor networks, algorithm design and image-processing.

Shahram Shirani (SM’04) received the B.Sc. degree in electrical engineering from the Isfahan University of Technology, Isfahan, Iran, in 1989, the M.Sc. degree in biomedical engineering from the Amirkabir University of Technology, Tehran, in 1994, and the Ph.D. degree in electrical and computer engineering from the University of British Columbia (UBC), Vancouver, BC, Canada, in 2000. Since July 2000, he has been with the department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada, where he is currently a Professor. His research interests include image and video compression, multimedia communications, and ultrasonic imaging. Dr. Shirani is a registered Professional Engineer (P. Eng.).