Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 828241, 12 pages
http://dx.doi.org/10.1155/2015/828241

Research Article

Simultaneous Pose and Correspondence Estimation Based on Genetic Algorithm

Haiwei Yang,[1] Fei Wang,[1] Zhe Li,[2] and Hang Dong[1]

[1] *Xi'an Jiaotong University, No. 28 Xianning West Road, Xi'an, Shaanxi 710049, China*
[2] *Xi'an Institute of Optics and Precision Mechanics, CAS, Xi'an, Shaanxi, China*

Correspondence should be addressed to Fei Wang; wfx@mail.xjtu.edu.cn

Received 28 February 2015; Revised 7 June 2015; Accepted 10 June 2015

Academic Editor: Chen Qian

Although several algorithms have been presented to solve the simultaneous pose and correspondence estimation problem, the correct solution may not be reached to with the traditional random-start initialization method. In this paper, we derive a novel method which estimates the initial value based on genetic algorithm, considering the influences of different initial guesses comprehensively. First, a set of random initial guesses is generated as candidate solutions. Second, the assignment matrix and the perspective projection error are computed for each candidate solution. And then each individual is modified (selection, crossover, and mutation) in current iterative process. Finally, the fittest individual is stochastically selected from the final population. With the presented initialization method, the proper initial guess could be first calculated and then the simultaneous pose and correspondence estimation problem could be solved easily. Simulation results with synthetic data and experiments on real images prove the effectiveness and robustness of our proposed method.

1. Introduction

Estimating the position and orientation using 2D images and known 3D targets, which is usually called pose estimation, is a basic problem in computer vision, including hand-eye coordination system, object tracking, augmented reality, and autonomous navigation [1–8]. However, when the correspondences between object features and image features are unknown, for example, because the scene contains repetitive patterns or because the 3D points are simply salient features on a geometric model without associated texture or because occlusion and clutter should be taken into account, it becomes difficult because no additional information is available.

In this case, pose and correspondence should be determined simultaneously, which is simultaneous pose and correspondence estimation problem. So far, designing a proper method that can be generally applied to estimate position and orientation when the correspondences are unknown is still a challenging problem. Particularly, this problem also takes occlusion, clutter, image noise, and model deformation into account. The problem can be iteratively solved by optimizing a global cost function. One limitation of such algorithm is that the global minimum cannot be guaranteed. This is alleviated by randomly initializing it at different initial guesses. Meanwhile, the algorithm is quite slow because hundreds of different initial poses are needed to try randomly and it may succeed only when the initial pose is close to the real pose. Obviously, the consideration of occlusion and clutter makes this problem complicated. Yet, certain configurations of the data or situations with large amounts of occlusion and clutter still cause the algorithm to fail.

In this paper, we formalize the simultaneous pose and correspondence problem comprehensively and present a novel initialization method based on genetic algorithm. We note that the relationships between each randomly initial guess are neglected by the traditional random-start initialization method. In this case, an initial guess may still be used to minimize the global objective function, even when a very similar guess has also been attempt. What is worse, an appropriate initial guess may not be selected in a long time if there are many local optima. Actually, each initial guess has different influence to the global optimum and a valid method should consider the different results of previous attempts during searching for the global optimum. Our initialization method is derived based on the analysis. Simulation results and experiments on real images suggest that our method has faster convergence speed and higher convergence rate than the traditional random-start method, which can be used to solve the simultaneous pose and correspondence problem effectively and robustly.

The rest of this paper is organized as follows: Section 2 carries out a study of simultaneous pose and correspondence method and some relevant proposals using the genetic algorithm. Section 3 provides a thorough analysis of the simultaneous pose and correspondence estimation problem and genetic algorithm. We describe our initialization method based on genetic algorithm in Section 4. Section 5 presents experimental results using both synthetic data and real images and compares our algorithm with the state-of-the-art pose estimation algorithms. Conclusions are drawn in Section 6.

2. Related Works

In literature, many algorithms have been developed to solve the simultaneous pose and correspondence problem. Most of them determine correspondences between 3D targets and 2D image features explicitly or implicitly before computing pose [9–11]. The hypothesize-and-test algorithm, for example, RANSAC, is frequently used. In this approach, a small set of 2D image features and 3D object features are selected first and the correspondences between them are then hypothesized. Based on the hypothesized correspondences, the pose can be easily computed. All the 3D object features are reprojected into the image plane with the estimated pose. If the original and reprojected images features are sufficiently similar, the pose is accepted; otherwise, a new hypothesis is formed and the process is repeated.

In the above algorithms, the correspondence problem and the pose estimation problem are solved separately, but the relationships between pose and correspondence are neglected. Time complexity may become great, especially when the number of 3D object points or 2D image points is high.

Skrypnyk and Lowe [12] presented another similar algorithm which uses view-variant 2D image features to index 3D models. In his approach, a process named off-line training is executed to learn 2D feature groupings associated with large number of different views of the 3D models. Then, the on-line recognition stage is performed to index 3D object models in a database of learned object-to-image correspondence hypotheses. Correspondences could be determined based on the object recognition results, which are used for pose estimation and final verification.

In [13], Wunsch and Hirzinger formalized the problem as the optimization of an objective function combining all the correspondence and pose constraints. Combined with a hybrid pose estimation algorithm, a random-start local search method is performed in the object-to-image space. However, in their algorithm, the correspondence constrains are not represented analytically. Instead, each object feature is explicitly matched to the closest sight line of the image features.

David et al. derived a very similar algorithm named SoftPOSIT in [14] by defining a global objective function, which combined an iterative correspondence assignment technique called Softassign [11] and an iterative pose estimation technique called POSIT [15] into a single iteration loop. SoftPOSIT stands out in the simultaneously estimating pose and correspondence approaches because of its accuracy and speed. The drawback is that it tries different initial poses and succeeds when the initial pose is close to the real pose.

SoftSI algorithm [16] proposed by Zhou et al. is also based on the global objective function, which can simultaneously obtain pose and correspondences. The SoftSI algorithm combines the SI algorithm and two singular value decomposition-(SVD-) based shape description theorems. By analyzing the calculation process of SI algorithm, the method can avoid pose ambiguity and quickly eliminate bad initial values.

This objective function can be also solved by different optimization algorithms [17–21] and can contains line features [22], cornerless images [23], and nonrigid shape [24]. Unfortunately, there is no guarantee of finding the global optimum given a single random-start initial guess in the algorithm, especially when the number of points is not equal because of occlusion, clutter, and image noise.

Genetic algorithm (GA), which simulated the natural evolution process, is a useful tool for many optimization problem. Recently, GA and its variations have been used in many fields, such as computer vision, artificial intelligence, pattern recognition, telecommunication, smart sensing, and mobile computing. In telecommunications, the quality of service (QoS) parameters may conflict with each other, so this problem is actually a multiobjective optimization [25]. A genetic algorithm can be used to solve complicated global optimization problems. In quantum computing, genetic algorithm can be used in stimulated annealing for combinatorial optimization so as to avoid premature convergence [26].

3. Problem Formulation

In this section, we first give a description of camera model and the formulation of pose estimation algorithm with known correspondences, using the closed-form global optimal function. Then the function is modified to characterize the global pose-correspondence problem without known correspondences. The genetic algorithm is introduced at last.

3.1. Camera Model. Generally, a camera can be seen as a pinhole model, which describes the mathematical relationship between the coordinates of a 3D point and its projection onto the image plane, where the camera aperture is described as a point. Some of the effects that the pinhole camera model does not take into account can be compensated, for example, by applying suitable coordinate transformations on the image coordinates. This means that the pinhole camera model often can be used as a reasonable description of how a camera depicts a 3D scene, for example, in computer vision and computer graphics.

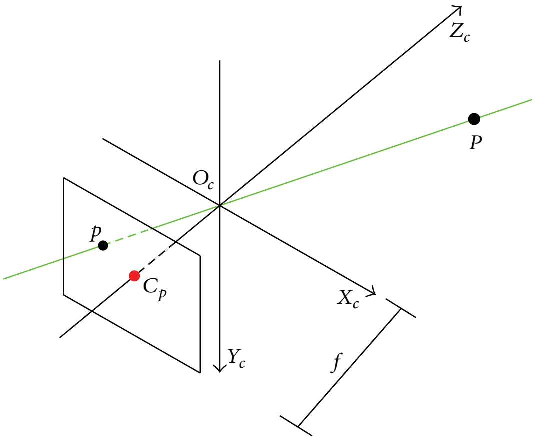


Figure 1: The pinhole model of a camera.

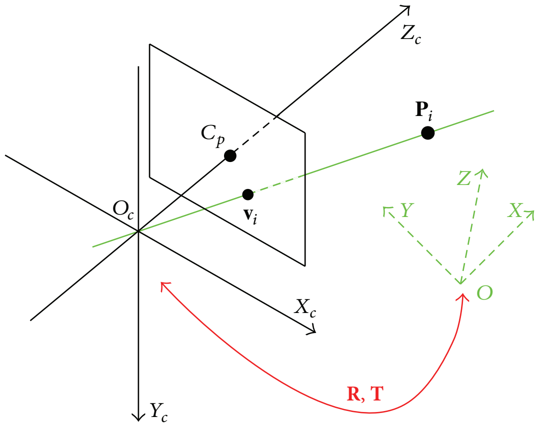


Figure 2: The perspective projection of a pinhole model.

The geometry related to the mapping of a pinhole camera is illustrated in Figure 1.

In the model, a 3D orthogonal coordinate system is abstracted with its origin at O_c, which is also where the camera aperture is located. The three axes of the coordinate system are referred to as X_c, Y_c, and Z_c. Axis Z_c is pointing in the viewing direction of the camera and is referred to as the optical axis, principal axis, or principal ray. The image plane is parallel to axes X_c and Y_c and is located at a distance f from the origin O_c in the negative direction of the Z_c axis. A practical implementation of a pinhole camera implies that the image plane is located such that it intersects Z_c axis at coordinate $-f$ where $f > 0$. f is also referred to as the focal length of the pinhole camera.

A point C_p at the intersection of the optical axis and the image plane is referred as the principal point or image center.

A point P somewhere in the world has the coordinates (X, Y, Z) relative to the axes X_c, Y_c, and Z_c. The projection line of point P into the camera (the green line in Figure 1) passes through point P and the point O_c. The projection of point P onto the image plane is denoted as p. This point is given by the intersection of the projection line (green line) and the image plane. In any practical situation we can assume that $Z > 0$, which means that the intersection point is well defined.

There is also a 2D coordinate system in the image plane, with origin at C_p and with axes X and Y which are parallel to X_c and Y_c, respectively. The coordinates of point p relative to this coordinate system is (x, y).

As depicted in Figure 1, the projection function can be formulated as

$$\begin{aligned} x &= f\frac{X}{Z}, \\ y &= f\frac{Y}{Z} \end{aligned} \tag{1}$$

which can be rewritten in homogenous coordinate system:

$$Z\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \tag{2}$$

In image coordinate system, there is an equation as follows:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & u_0 \\ 0 & s_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{3}$$

where u_0 and v_0 are the image center in image coordinate system and s_x and s_y are scale factors.

According to (2) and (3), the pinhole model of a camera can be described as

$$\begin{aligned} u - u_0 &= fs_x\frac{X}{Z} = f_x\frac{X}{Z}, \\ v - v_0 &= fs_y\frac{Y}{Z} = f_y\frac{Y}{Z}, \end{aligned} \tag{4}$$

where $f_x = fs_x$ and $f_y = fs_y$ are defined as equivalent focal length of axes X_c and Y_c, respectively. f_x, f_y, u_0, and v_0 are called the camera intrinsic parameters.

3.2. Pose Estimation with Known Correspondences. The mapping from 3D to 2D coordinates described by a pinhole camera is a perspective projection followed by a 180° rotation in the image plane, which corresponds to how a real pinhole camera operates. The resulting image is rotated 180° and the relative size of projected objects depends on their distance to the focal point and the overall size of the image depends on the distance f between the image plane and the focal point. In order to produce an unrotated image, which is what we expect from a camera, we can place the image plane so that it intersects the Z_c axis at f instead of at $-f$ and rework the previous calculations. This would generate a virtual (or front) image plane which cannot be implemented in practice but provides a theoretical camera which may be simpler to analyze than the real one. This relationship is illuminated in Figure 2.

Assuming that the coordinates of corresponding 3D object points and normalized 2D image points are

$\mathbf{P}_i = [X_i, Y_i, Z_i]^T$ and $\mathbf{v}_i = [x_i, y_i]^T$, the perspective projection camera model can be described as

$$l_i \begin{bmatrix} \mathbf{v}_i \\ 1 \end{bmatrix} = \mathbf{R}\mathbf{P}_i + \mathbf{T}, \tag{5}$$

where l_i is a scale factor. $\mathbf{R}$ and $\mathbf{T}$ are the rotation matrix and translation vector, respectively, which describe the relationship between the camera and the 3D target. Pose estimation with known correspondences is then to seek the optimal rotation matrix $\mathbf{R}$ and translation vector $\mathbf{T}$ that best satisfy the equations in (5). According to the above equation, $\mathbf{R}$ is a 3×3 matrix and $\mathbf{T}$ is a 3×1 vector, which can described as

$$\begin{gathered} \mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1^T \\ \mathbf{R}_2^T \\ \mathbf{R}_3^T \end{bmatrix}, \\ \mathbf{T} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}, \end{gathered} \tag{6}$$

where $\mathbf{R}_1^T$, $\mathbf{R}_2^T$, and $\mathbf{R}_3^T$ indicate the row vectors of rotation matrix $\mathbf{R}$ and each of them is a unit vector. t_x, t_y, and t_z represent the coordinates of the original point of object coordinate system onto the three axes of the camera.

The perspective projection camera model can then be rewritten as

$$\begin{bmatrix} l_i x_i \\ l_i y_i \\ l_i \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1^T & t_x \\ \mathbf{R}_2^T & t_y \\ \mathbf{R}_3^T & t_z \end{bmatrix} \begin{bmatrix} \mathbf{P}_i \\ 1 \end{bmatrix}. \tag{7}$$

If we multiply the same factor $s = 1/t_z$ on all the elements of the right-side perspective projection matrix and the left-side homogeneous image point coordinates, the equality is not affected. Introducing a scaling factor $w_i = l_i/t_z$, we obtain

$$\begin{bmatrix} w_i x_i \\ w_i y_i \end{bmatrix} = \begin{bmatrix} s\mathbf{R}_1^T & st_x \\ s\mathbf{R}_2^T & st_y \end{bmatrix} \begin{bmatrix} \mathbf{P}_i \\ 1 \end{bmatrix} \tag{8}$$

with

$$w_i = \mathbf{R}_3 \cdot \frac{\mathbf{P}_i}{t_z} + 1. \tag{9}$$

In photogrammetry, the pose estimation problem is usually formulated as the problem of optimizing the following objective function:

$$E = \sum_{i=1}^{N} \left(\left(\mathbf{Q}_x \cdot \mathbf{P}_i - w_i x_i \right)^2 + \left(\mathbf{Q}_y \cdot \mathbf{P}_i - w_i y_i \right)^2 \right), \tag{10}$$

where N is the number of 2D image points. To simplify the subsequent notation, we introduce three new vectors $\mathbf{Q}_x$, $\mathbf{Q}_y$, and $\widetilde{\mathbf{P}}_i$ with four homogeneous coordinates:

$$\begin{gathered} \mathbf{Q}_x = s\left(\mathbf{R}_1, t_x\right), \\ \mathbf{Q}_y = s\left(\mathbf{R}_2, t_y\right), \\ \widetilde{\mathbf{P}}_i = \left(\mathbf{P}_i, 1\right). \end{gathered} \tag{11}$$

We call $\mathbf{Q}_x$ and $\mathbf{Q}_y$ the pose vectors.

In the objective function E, the pose vectors are those for which all the partial derivatives of the objective function with respect to the coordinates of these vectors are zero. $\mathbf{Q}_x$ and $\mathbf{Q}_y$ can then be solved using the conditions:

$$\begin{gathered} \mathbf{Q}_x = \left(\sum_{i=1}^{N} \mathbf{P}_i \mathbf{P}_i^T \right)^{-1} \left(\sum_{i=1}^{N} w_i x_i \mathbf{P}_i \right), \\ \mathbf{Q}_y = \left(\sum_{i=1}^{N} \mathbf{P}_i \mathbf{P}_i^T \right)^{-1} \left(\sum_{i=1}^{N} w_i y_i \mathbf{P}_i \right). \end{gathered} \tag{12}$$

3.3. Pose Estimation with Unknown Correspondences. According to the above analysis with known correspondences, pose vectors can be computed in each iterative loop by minimizing the object function E. When correspondences are unknown, each image point $\mathbf{v}_i$ can potentially match any of the object points $\mathbf{P}_j$, and therefore the scaling factor w should be corrected specific to the coordinates of $\mathbf{P}_j$:

$$w_j = \mathbf{R}_3 \cdot \frac{\mathbf{P}_j}{t_z} + 1, \quad j = 1, 2, \ldots, M, \tag{13}$$

where M is the number of 3D object points.

Therefore, the simultaneous pose and correspondence problem can then be formulated as follows:

$$\begin{aligned} &E \\ &\quad = \sum_{i=1}^{N} \sum_{j=1}^{M} m_{ij} \left(\left(\mathbf{Q}_x \cdot \mathbf{P}_i - w_i x_i \right)^2 + \left(\mathbf{Q}_y \cdot \mathbf{P}_i - w_i y_i \right)^2 \right), \end{aligned} \tag{14}$$

where m_{ij} (equal to 0 or 1) are correspondence weights.

In next stage, we should find a zero-one assignment matrix $\mathbf{m} = \{m_{ij}\}$, which specifies the matching between a set of N image points and a set of M object points explicitly. The assignment matrix $\mathbf{m}$, which has one row for each of image point $\mathbf{v}_i$ and one column for each of object point, must satisfy the constraint that each 2D image point matches at most one 3D object point and vice versa. Given that pose vectors have been estimated, the assignment matrix can be calculated by the iterative Softassign technique [11].

Assuming the correspondence variables m_{ij} are known and fixed, the pose vectors $\mathbf{Q}_x$ and $\mathbf{Q}_y$ can then be determined through minimizing the objective function in an iteration step. The solutions are

$$\mathbf{Q}_x = \left(\sum_{j=1}^{M} m'_j \mathbf{P}_j \mathbf{P}_j^T\right)^{-1} \left(\sum_{i=1}^{N}\sum_{j=1}^{M} m_{ij} w_j x_i \mathbf{P}_j\right),$$
$$\mathbf{Q}_y = \left(\sum_{j=1}^{M} m'_j \mathbf{P}_j \mathbf{P}_j^T\right)^{-1} \left(\sum_{i=1}^{N}\sum_{j=1}^{M} m_{ij} w_j y_i \mathbf{P}_j\right), \quad (15)$$

with $m'_k = \sum_{i=1}^{N} m_{ij}$.

The objective function is minimized iteratively, with the following three operations at each iteration step:

(1) Given the initial guess of pose vectors $\mathbf{Q}_x$ and $\mathbf{Q}_y$, compute the assignment matrix $\mathbf{m}$.

(2) Assuming the scaling factor w_j is known and fixed, calculate the pose vectors using the assignment matrix $\mathbf{m}$.

(3) Using the estimated pose vectors $\mathbf{Q}_x$ and $\mathbf{Q}_y$, update the scaling factor w_j.

The above iterative approach can be summarized as follows. First, given an appropriate initial guess for the pose vectors according to the set of 2D image points and 3D object points, then the assignment matrix which represents correspondence between the image points and object points is estimated. Finally, the pose vectors are updated, using the results of the correspondences between 2D image points and 3D object points. This process is repeated until these estimations converge. In this case, through the final pose vectors

$$\mathbf{Q}_x = \left(q_{x1}, q_{x2}, q_{x3}, q_{x4}\right)^T,$$
$$\mathbf{Q}_y = \left(q_{y1}, q_{y2}, q_{y3}, q_{y4}\right)^T, \quad (16)$$

pose variables can be calculated as follows:

$$s = \sqrt{\left\|\left(q_{x1}, q_{x2}, q_{x3}\right)\right\| \cdot \left\|\left(q_{y1}, q_{y2}, q_{y3}\right)\right\|},$$
$$\mathbf{R}_1 = \frac{\left(q_{x1}, q_{x2}, q_{x3}\right)^T}{s},$$
$$\mathbf{R}_2 = \frac{\left(q_{y1}, q_{y2}, q_{y3}\right)^T}{s},$$
$$\mathbf{R}_3 = \mathbf{R}_1 \times \mathbf{R}_2, \quad (17)$$
$$t_x = \frac{q_{x4}}{s},$$
$$t_y = \frac{q_{y4}}{s},$$
$$t_z = \frac{1}{s}.$$

Then the final rotation matrix $\mathbf{R}$ and translation vector $\mathbf{T}$ are

$$\mathbf{R} = \left[\mathbf{R}_1 \;\; \mathbf{R}_2 \;\; \mathbf{R}_3\right]^T,$$
$$\mathbf{T} = \left(t_x, t_y, t_z\right)^T. \quad (18)$$

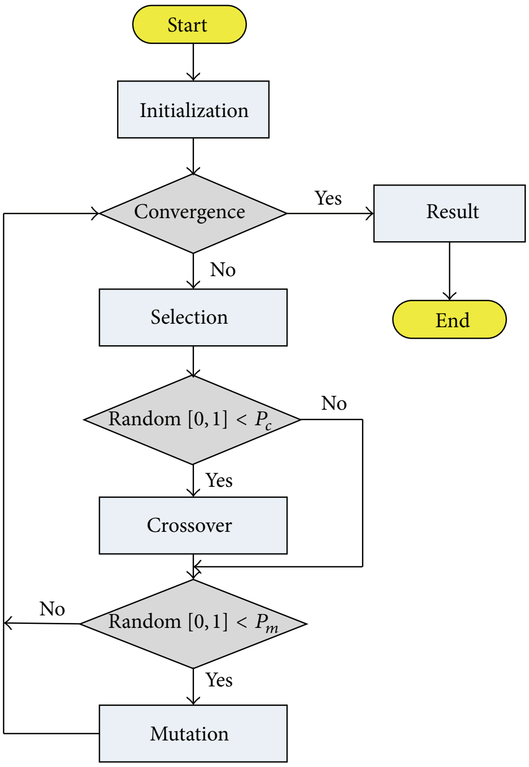


Figure 3: The flow chart of genetic algorithm.

3.4. Genetic Algorithm.

3.4. Genetic Algorithm. Genetic algorithm (GA) was an optimal model which simulated the natural evolution process [25–27]. The algorithm was first presented by John Holland in 1975.

In a genetic algorithm, a population of candidate solutions, called individuals, to an optimization problem is evolved toward better solutions. Each candidate solution has a set of properties, its chromosomes, which can be mutated and altered. Traditionally, candidate solutions are represented in binary as strings of 0 and 1, but other encodings are also possible.

The evolution usually starts from a population of randomly generated individuals and is an iterative process, and the population in each iteration is called a generation. In each generation, the fitness of every individual in the population is evaluated, which is usually the value of the objective function in the optimization problem being solved. The more fit individuals are stochastically selected from the current population, and each individual's genome is modified to form a new generation. The new generation of candidate solutions is then used in the next iteration of the genetic algorithm. Commonly, the genetic algorithm terminates when either a maximum number of generations has been produced or a satisfactory fitness level has been reached for the population.

A typical genetic algorithm requires two aspects. The first one is a genetic representation of the solution domain, and the second one is a fitness function to evaluate the solution domain [25–27].

The flow chart of genetic algorithm is depicted in Figure 3, where p_c is the probability of crossover and the p_m is the probability of mutation.

Once the genetic representation and the fitness function are defined, a GA proceeds to initialize a population of solutions and then to improve it through repetitive application of the mutation, crossover, inversion, and selection operators.

4. Our Initialization Method

To solve the simultaneous pose and correspondence problem, an iterative process starting from an initial guess is necessary [14]. Usually, the minimized objective function has many local optimal because of occlusion, clutter, and image noise. If the initial guess is not appropriate, the correct global optimum will not be reached to or the objective function will converge to its other local optimal. Therefore we present an effective strategy based on genetic algorithm to solve the simultaneous pose and correspondence problem.

4.1. Initialization. The random-start method is a common initialization method, which starts from a random initial guess. There are 12 different variables, including a 3 × 3 rotation matrix **R** and a 3×1 translation vector **T**. All the variables can be generated through six degrees of freedom and include three Euler angles and the translation parameters. Therefore the traditional random-start initialization method generates a random 6D vectors $p_0 = (\alpha, \beta, \gamma, x, y, z)$ in a 6D hypercube. Then the rotation matrix **R** and translation vector **T** are defined as

$$\mathbf{R} = \begin{bmatrix} \cos\gamma\cos\beta & -\sin\gamma\cos\alpha + \cos\gamma\sin\beta\sin\alpha & \sin\gamma\sin\alpha + \cos\gamma\sin\beta\cos\alpha \\ \sin\gamma\cos\beta & \cos\gamma\cos\alpha + \sin\gamma\sin\beta\sin\alpha & -\cos\gamma\sin\alpha + \sin\gamma\sin\beta\cos\alpha \\ -\sin\beta & \cos\beta\sin\alpha & \cos\beta\cos\alpha \end{bmatrix}, \quad \mathbf{T} = (x, y, z)^T. \tag{19}$$

4.2. Algorithm Framework. In the traditional random-start initialization method, beginning from a random initial guess, pose and correspondence are iteratively estimated until meeting a specified termination criterion. If the calculated pose and correspondence are not correct, all the steps are processed again with another new random initial guess. However, the random-start method does not consider the different influence of each random initial guess because they are independent with each other in this method. Even worse, the initial guess p_0 may repeat in some range, and the count of initial guess will increase rapidly and the convergence rate will decrease sharply.

To get an appropriate initial guess, we present a new initialization method based on genetic algorithm [21]. First, instead of generating an initial guess one by one, a set of K random initial guesses $\{p_0^{(k)}\}$, $k = 1, 2, \ldots, K$, is generated as candidate solutions called a population. Second, the assignment matrix **m** is computed for each candidate solution in the population and then the perspective projection error E is evaluated as the fitness of every individual in the population. Finally, the more fit individuals are stochastically selected from the current population, and each individual is modified (selection, crossover, and mutation) to form a new population. The new population of candidate solutions is then used in the next iteration of the algorithm.

To illustrate the three modified operations, we use two candidate solutions in an iterative step as example:

$$p_0^{(i)} = \left(\alpha^{(i)}, \beta^{(i)}, \gamma^{(i)}, x^{(i)}, y^{(i)}, z^{(i)}\right), \quad p_0^{(j)} = \left(\alpha^{(j)}, \beta^{(j)}, \gamma^{(j)}, x^{(j)}, y^{(j)}, z^{(j)}\right). \tag{20}$$

We define three genetic algorithm operations.

Selection. Consider

$$p_0^{(i+1)} = \left(\alpha^{(i)}, \beta^{(i)}, \gamma^{(i)}, x^{(i)}, y^{(i)}, z^{(i)}\right), \quad p_0^{(j+1)} = \left(\alpha^{(j)}, \beta^{(j)}, \gamma^{(j)}, x^{(j)}, y^{(j)}, z^{(j)}\right). \tag{21}$$

Crossover. Consider

$$p_0^{(i+1)} = \left(\alpha^{(i)}, \beta^{(i)}, \gamma^{(i)}, x^{(j)}, y^{(j)}, z^{(j)}\right), \quad p_0^{(j+1)} = \left(\alpha^{(j)}, \beta^{(j)}, \gamma^{(j)}, x^{(i)}, y^{(i)}, z^{(i)}\right). \tag{22}$$

Mutation. Consider

$$p_0^{(i+1)} = \left(\alpha', \beta', \gamma', x^{(i)}, y^{(i)}, z^{(i)}\right), \quad p_0^{(j+1)} = \left(\alpha'', \beta'', \gamma'', x^{(j)}, y^{(j)}, z^{(j)}\right). \tag{23}$$

Commonly, the iterative process will terminate when either a maximum number of iterations has been produced or a satisfactory fitness level has been reached for the population.

In practice, the search for better solution should be terminated when the current solution is such that the number of matching points is smaller than the threshold t_m, which is defined as

$$t_m = \rho N, \quad 0 < \rho \le 1, \tag{24}$$

where N is the total number of the image points and ρ is a scale factor of matching rate, which determines what percent of the detected object points must be matched.

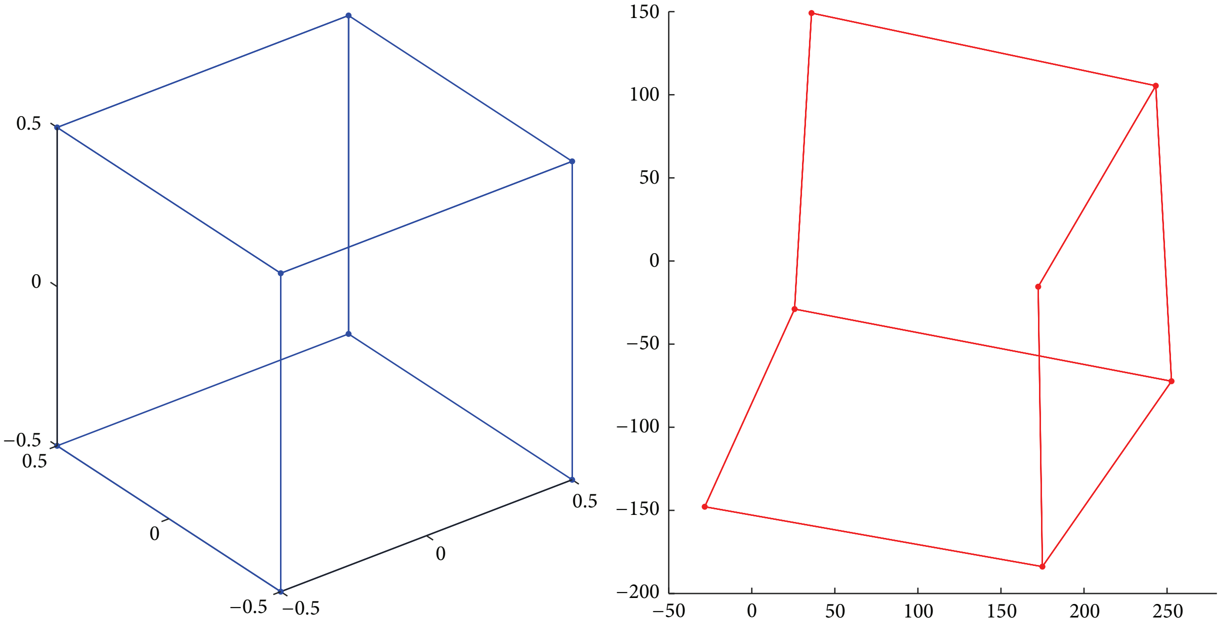


Figure 4: 3D object points and 2D image points with one occluded point.

In this case, we can deduce that the current initial guess is not suitable for the simultaneous pose and correspondence problem and a new initial guess should be given. Due to image noise, occlusion, and measurement error, ρ will not reach to 1 even when a good initial guess is found. Therefore, in the experiments discussed below, we take $\rho \equiv 0.75$.

This test is not perfect, as it is possible for an initial guess to be very accurate even when the number of matched points is less than this threshold, which occurs mainly in cases of high noise. Conversely, a wrong initial guess may be accepted when the ratio of clutter features to detected object points is high. However, these situations are relatively uncommon both in simulation case and in practice.

5. Experiments

We first show the simulation results to confirm the effectiveness of our initialization method, compared with traditional random-start initialization method. Then experiments with real images intuitively present the computed pose and correspondences.

5.1. Synthetic Experiments. We generate synthetic data as follows: 3D object has eight points, distributing in the vertexes of a cube; there are seven 2D image points with unknown correspondence to 3D object points, where one vertex of the cube is occluded, as shown in Figure 4. All the coordinates of image points have been normalized for the sake of convenience. Connecting edges are used for understanding, instead of computing.

In synthetic experiments, the convergence rate and convergence speed of global object function are evaluated, respectively, initialized with the traditional random-start method and our novel initial method.

An initial guess is convergent only when the global objective function E in (14) reaches to a stable minimum with finite iterations. It means that, with the initial guess, pose and correspondence can be calculated simultaneously in a few of iterations.

We first compute the convergence rate (CR) at different noise levels, which is defined as

$$\mathrm{CR} = \frac{\text{number of convergent initial guesses}}{\text{total number of initial guesses}}, \qquad (25)$$

where total number of initial guesses is set to 1000 in each different noise level.

Figure 5 shows the convergence rates of our initial method and the traditional random start method as a function of noise level. In this figure, each point depicts results averaged over 1000 random initial guesses. It can be seen that our initial method has a higher convergence rate and decreases more slowly than the traditional random-start method. This is due to the fact that our initial method using genetic algorithm considers different influences between each initial guess and has a higher probability of approaching the minimum of the global objective function than the traditional random start method.

In the synthetic experiment, a set of random initial guesses is first generated as candidate solutions in our method based on genetic algorithm. Then certain processes are done with all the initial guesses until that the proper pose is determined. Similarly, the traditional random-start method also generates a list of globally aware guesses and tests against these random guesses as a whole. At this time, if the list of initial guesses contains the proper pose, the selection process terminates. However, the proper pose may be not in the list, and in this case, another list of initial guesses is needed to randomly generate. The whole procedure does not consider

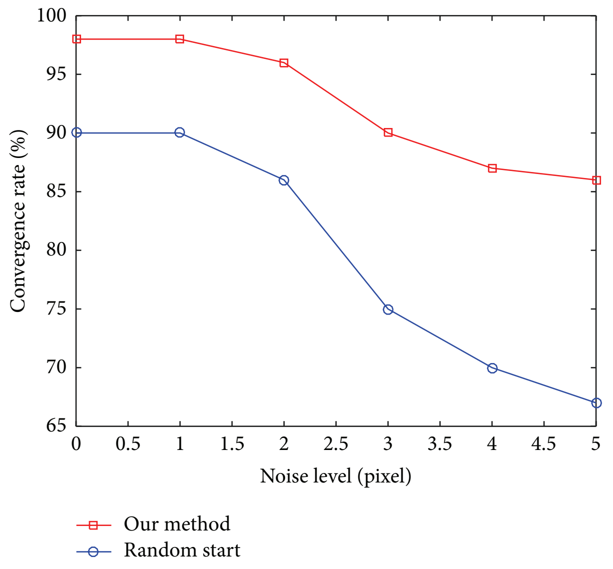


FIGURE 5: Convergence rate at different noise level. The results initialized with random start method are plotted as circles (○) and the results using our initialized method are plotted as squares (□).

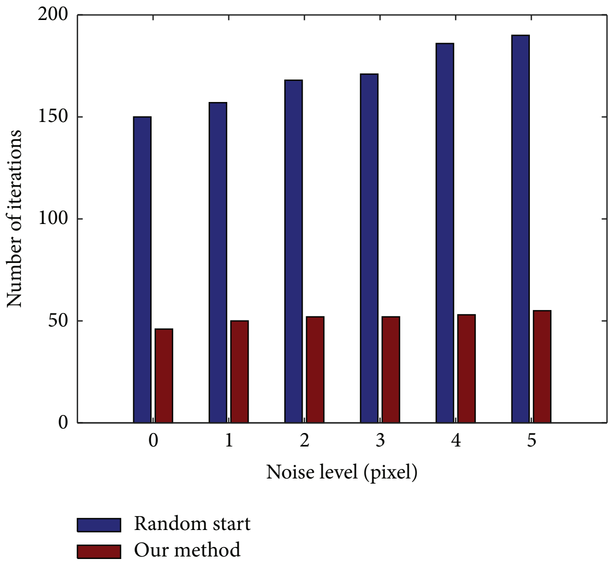


FIGURE 7: Number of iterations at different noise level.

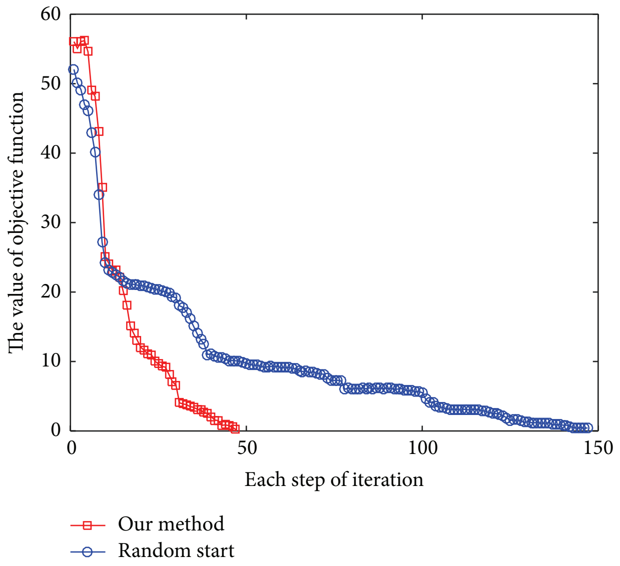


FIGURE 6: The value of object function E at each iteration without noise. The results initialized with random start method are plotted as circles (○) and the results using our initialized method are plotted as squares (□).

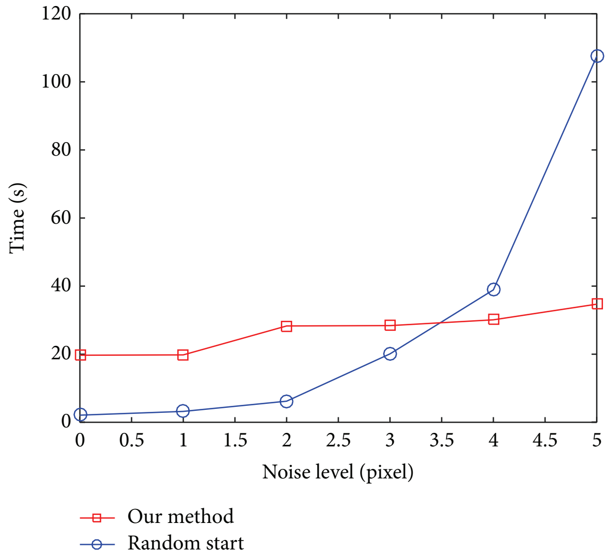


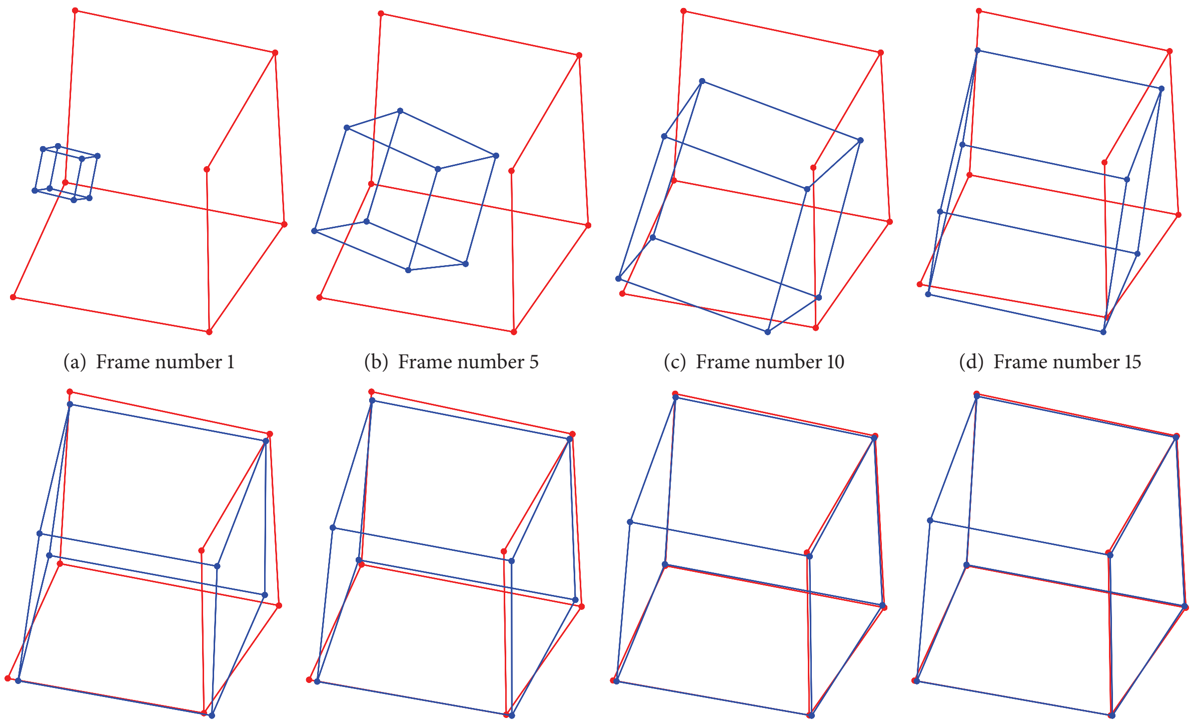
FIGURE 8: Average time consumption at different noise level.

the former initial guess that has already been computed. On the opposite, our method based on genetic algorithm takes all the initial guesses as a whole population and new initial guess is generated according the results of former population.

We then compare the convergence speed when an initial guess computed by our method or random start method is convergent. The results are depicted in Figures 6 and 7.

From Figures 6 and 7, we can see that the value of objective function E decreases quickly with our initial guess and the number of iterations is always low. However, initialized with random start, the value of object function E decreases quickly only at the first several iterations and becomes very slowly at the subsequent iterations, which leads to a very large number of iterations. Meanwhile, the random start method does not consider the different influence of each random initial guess because every initial guess is independent with each other. Even worse, the initial guess may repeat in some range, and the count of iterations will increase rapidly and the convergence speed will decrease greatly.

When in presence of large amounts of clutter, occlusions, or image noise, the random-start method searches for the proper initial pose in the pose space by using a RANSAC-style approach. It should consider all possible correspondences between 2D image points and 3D object points. What is worse, the initial guesses are represented in a 6D pose space


(a) Frame number 1 (b) Frame number 5 (c) Frame number 10 (d) Frame number 15

(e) Frame number 20 (f) Frame number 25 (g) Frame number 30 (h) Frame number 38

FIGURE 9: The iteration step of perspective projection with an initial value calculated by our method.

($p_0 = (\alpha, \beta, \gamma, x, y, z)$). As a consequence, it would lead to computational explosion and quickly become computationally intractable with realistic numbers of features.

Figure 8 depicts the average time consumption until the initial guess is convergence.

When the noise level is low, the initial guess in the 6D pose space has high probability to converge. Only a few of times is needed to determine the proper guess with the randomly selected initial pose. However, when the noise level becomes high, the time consumption increases obviously because the convergence rate fast decreases and more times are needed to try to find a proper initial value. On the opposite, the average time consumption increases slowly with the different noise level.

Figure 9 shows an example computation of simultaneous pose and correspondence algorithm initialized with our method, using the cube model depicted in Figure 4.

In Figure 9, we can see that the perspective projection of 3D object points overlaps with original 2D image points gradually after few iterations, giving the appreciate initial guess calculated by our method, which means that pose between the coordinate system of 3D object and the coordinate system of camera can be estimated with unknown correspondences and occlusion.

5.2. Real Images. We test our initialization method for simultaneous pose and correspondence problem on real images.

In the test, there are 11 or 10 image points in 2D image and 28 object points in the 3D model. An initial pose is first given by random start method (the results are showed in Figure 10) and then by our initialization method (the results are showed in Figure 11). Figure 12 is another experiment results initialized using our method.

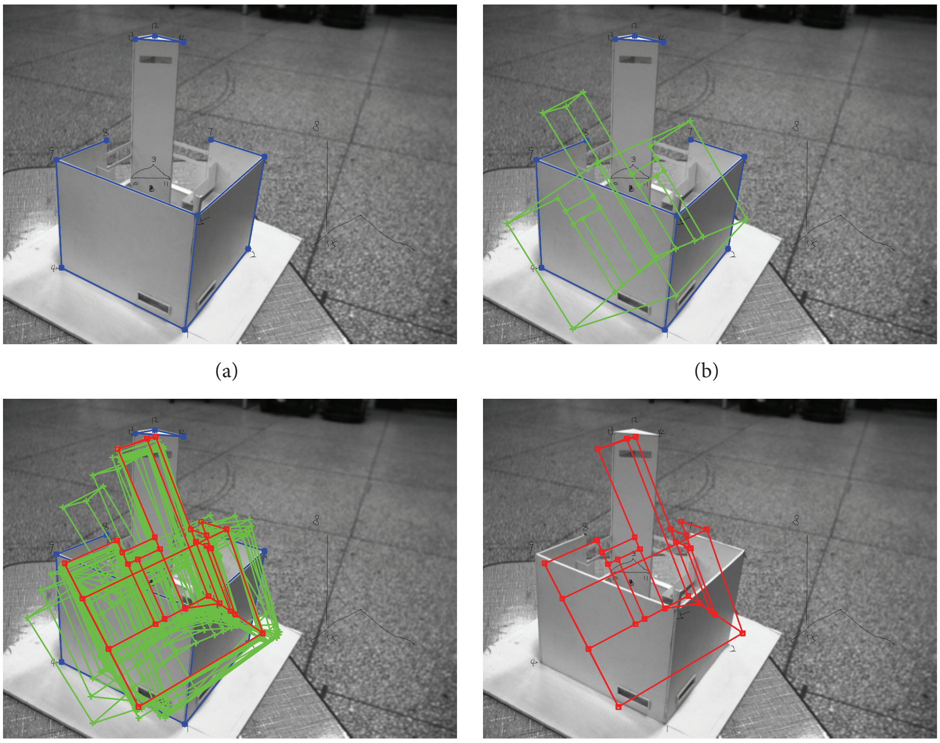
3D object model with 28 points is known and 2D image points are recognized by our implementation of Harris algorithm [12].

As depicted in Figure 10, with the unsuitable initial pose given by random start method, the final pose is not correct and the simultaneous pose and correspondence problem cannot be solved very well.

As shown in Figures 11 and 12, the number of image points is less than 3D object points; what is more, the correspondences between 2D image points and 3D object points are unknown due to occlusion, clutter, and image noise. Using our initialization method based on genetic algorithm, an appropriate initial guess is first calculated and then pose and correspondences are estimated with the initial value. Finally, 3D object model is accurately reprojected onto image plane using the rotation matrix and translation vector contained in pose. The effectiveness of the initial guess can be verified by observing the overlapping degree between the perspective projection and the original image.

6. Conclusions and Future Work

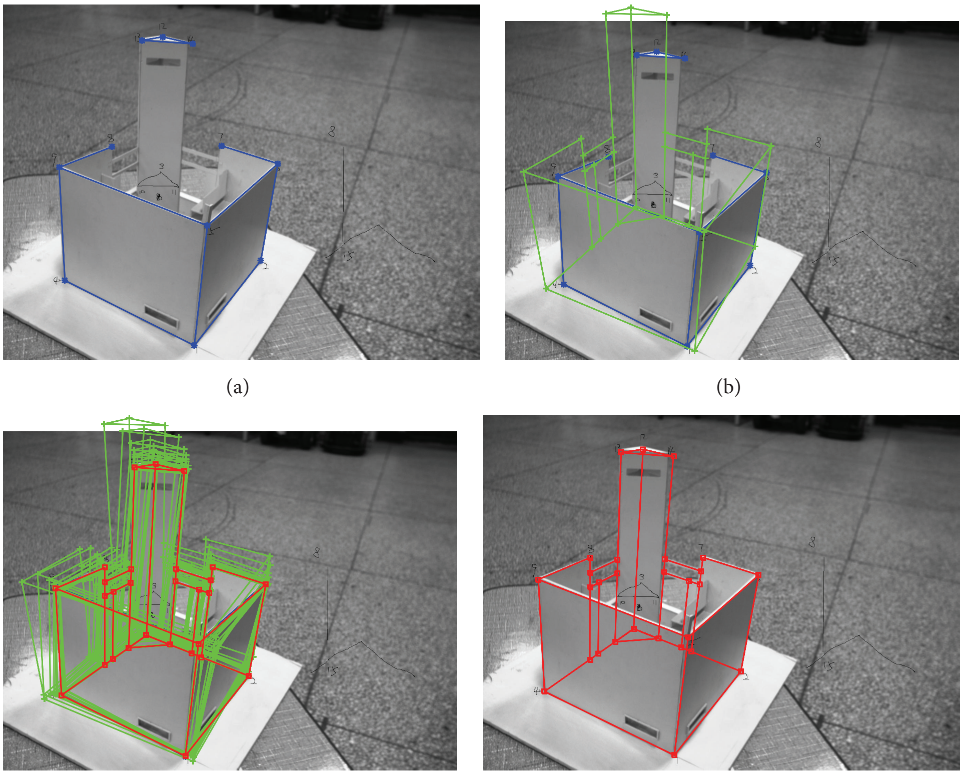
In this paper, we present an initialization method for determining the pose of 3D objects from images to solve the simultaneous pose and correspondence problem. Pose and correspondences can be computed simultaneously by minimizing



(a) (b)

(c) (d)

Figure 10: The results of real image initialized by random start method (11 image points): (a) 2D image points s (point number 9); (b) the projection of 3D model with the calculated initial pose; (c) the projection results at each iteration; (d) the final result of simultaneous pose and correspondence algorithm.



(a) (b)

(c) (d)

Figure 11: The results of real image initialized by our method (11 image points): (a) 2D image points; (b) the projection of 3D model with the calculated initial pose; (c) the projection results at each iteration; (d) the final result of simultaneous pose and correspondence algorithm.

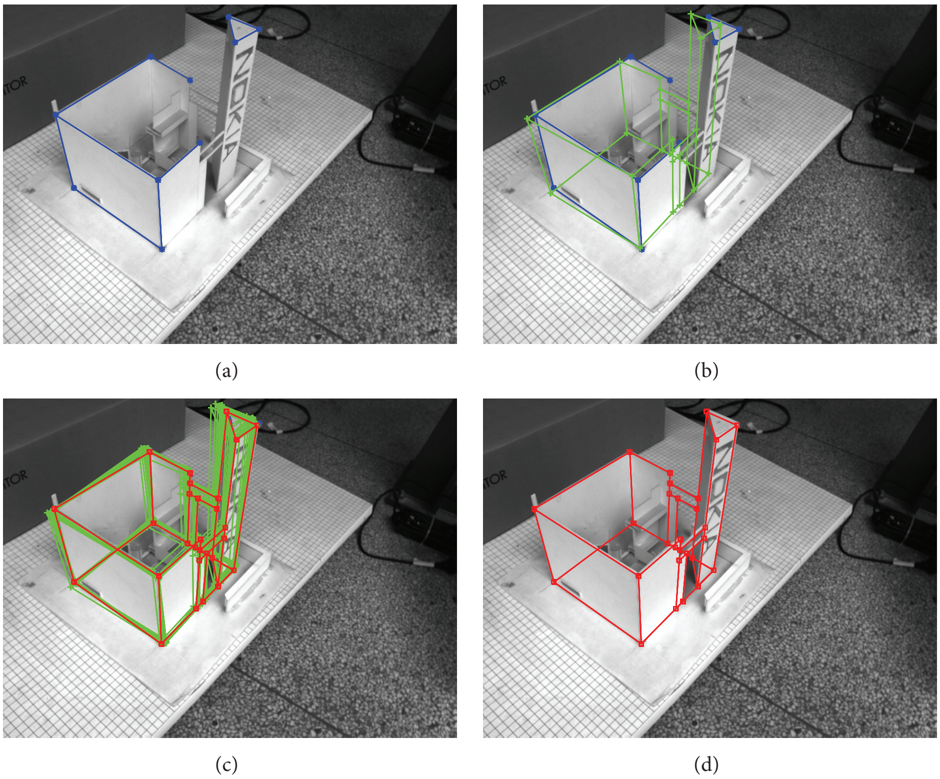


Figure 12: The results of real image initialized by our method (10 image points): (a) 2D image points; (b) the projection of 3D model with the calculated initial pose; (c) the projection results at each iteration; (d) the final result of simultaneous pose and correspondence algorithm.

a global objective function initialized using our initial pose, which is calculated based on genetic algorithm considering the influences of different initial values comprehensively. Compared with the traditional random-start initialization method, our proposed method has higher convergence rate and lower number of iterations, which has been verified through experiments.

Future work will involve initializing the simultaneous pose and correspondence algorithm automatically using special features extracted in 3D object and 2D image plane. The effectiveness of other optimization algorithms would be analyzed. We are also interested in implementing a more thorough formalism to include initialization, pose estimation, and correspondence determination.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the Fund of National Natural Science Foundation of China (61231018) and National High Technology Research and Development Program of China (2013AA014601).

References

[1] A. Ansar and K. Daniilidis, "Linear pose estimation from points or lines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 578–589, 2003.

[2] J. A. Hesch and S. I. Roumeliotis, "A direct least-squares (dls) solution for PnP," in *Proceedings of the International Conference on Computer Vision*, Barcelona, Spain, November 2011.

[3] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 930–943, 2003.

[4] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: an accurate O(n) solution to the PnP problem," *International Journal of Computer Vision*, vol. 81, no. 2, pp. 155–166, 2009.

[5] P. D. Fiore, "Efficient linear solution of exterior orientation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 140–148, 2001.

[6] G. Schweighofer and A. Pinz, "Globally optimal O(n) solution to the PnP problem for general camera models," in *Proceedings of the 19th British Machine Vision Conference (BMVC' 08)*, pp. 1–10, Leeds, UK, September 2008.

[7] S. Malik, G. Roth, and C. McDonald, "Robust 2D tracking for real-time augmented reality," in *Proceedings of the Conference on Vision Interface*, 2002.

[8] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the Association for Computing Machinery*, vol. 24, no. 6, pp. 381–395, 1981.

[9] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjolsness, "New algorithms for 2D and 3D point matching: pose estimation and correspondence," *Pattern Recognition*, vol. 31, no. 8, pp. 1019–1031, 1998.

[10] D. DeMenthon, P. David, and H. Samet, "SoftPOSIT: an algorithm for registration of 3D models to noisy perspective images combining softassign and POSIT," Tech. Rep. CS-TR-969, CS-TR 4257, University of Maryland, College Park, Md, USA, 2001.

[11] S. Gold and A. Rangarajan, "A graduated assignment algorithm for graph matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 4, pp. 377–388, 1996.

[12] I. Skrypnyk and D. G. Lowe, "Scene modelling, recognition and tracking with invariant image features," in *Proceedings of the 3rd IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '04)*, pp. 110–119, IEEE, Arlington, Va, USA, November 2004.

[13] P. Wunsch and G. Hirzinger, "Registration of CAD-models to images by iterative inverse perspective matching," in *Proceedings of the 13th International Conference on Pattern Recognition (ICPR '96)*, vol. 1, pp. 78–83, August 1996.

[14] P. David, D. Dementhon, R. Duraiswami, and H. Samet, "SoftPOSIT: simultaneous pose and correspondence determination," *International Journal of Computer Vision*, vol. 59, no. 3, pp. 259–284, 2004.

[15] D. Oberkampf, D. F. DeMenthon, and L. S. Davis, "Iterative pose estimation using Coplanar feature points," *Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 495–511, 1996.

[16] H. Zhou, T. Zhang, and W. Lu, "Vision-based pose estimation from points with unknown correspondences," *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3468–3477, 2014.

[17] V. Lepetit and P. Fua, "Keypoint recognition using randomized trees," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1465–1479, 2006.

[18] W.-Y. Lin, L.-F. Cheong, P. Tan, G. Dong, and S. Liu, "Simultaneous camera pose and correspondence estimation with motion coherence," *International Journal of Computer Vision*, vol. 96, no. 2, pp. 145–161, 2012.

[19] B. Raytchev and Y. Kimura, "Real-time 3D pose and correspondence from stereo image sequences by combinatorial optimization," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2012)*, pp. 1–8, IEEE, Brisbane, Australia, June 2012.

[20] J. Xia, X. Xu, and J. Xiong, "Simultaneous pose and correspondence determination using Differential Evolution," in *Proceedings of the 8th International Conference on Natural Computation (ICNC '12)*, pp. 703–707, May 2012.

[21] H. Yang, F. Wang, Y. Song, and L. Chen, "A novel initialization method based on genetic algorithm for simultaneous pose and correspondence estimation," in *Proceedings of the 4th International Conference on the Innovative Computing Technology*, 2014.

[22] P. David, D. DeMenthon, R. Duraiswami, and H. Samet, "Simultaneous pose and correspondence determination using line features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 424–431, June 2003.

[23] W.-Y. Lin, G. Dong, P. Tan, L.-F. Cheong, and C.-H. Yan, "Simultaneous camera pose and correspondence estimation in cornerless images," in *Proceedings of the IEEE 12th International Conference on Computer Vision*, pp. 1179–1186, 2009.

[24] J. Śanchez-Riera, J. Östlund, P. Fua, and F. Moreno-Noguer, "Simultaneous pose, correspondence and non-rigid shape," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1189–1196, June 2010.

[25] S. Dong and W. Dong, "A QoS driven web service composition method based on ESGA (Elitist Selection Genetic Algorithm) with an improved initial population selection strategy," *International Journal of Distributed Sensor Networks*, vol. 5, no. 1, p. 54, 2009.

[26] W. Shu, "Quantum-inspired genetic algorithm based on simulated annealing for combinatorial optimization problem," *International Journal of Distributed Sensor Networks*, vol. 5, no. 1, pp. 64–65, 2009.

[27] R. Akbari and K. Ziarati, "A multilevel evolutionary algorithm for optimizing numerical functions," *International Journal of Industrial Engineering Computations*, vol. 2, no. 2, pp. 419–430, 2011.

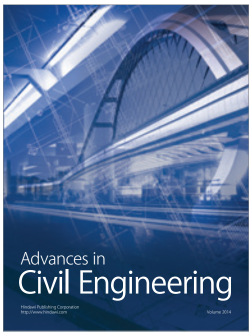
Journal of
Engineering

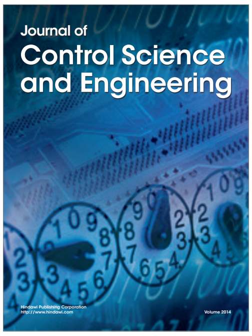
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

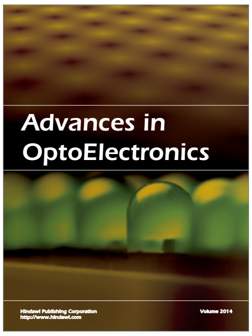
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

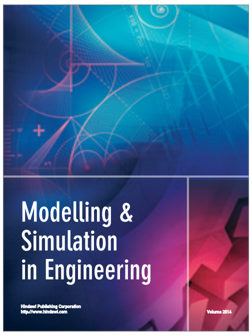
Journal of
Robotics

Journal of
Electrical and Computer Engineering

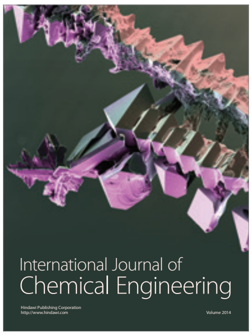

Advances in
OptoElectronics

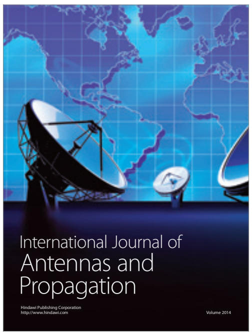
VLSI Design

International Journal of
Navigation and Observation

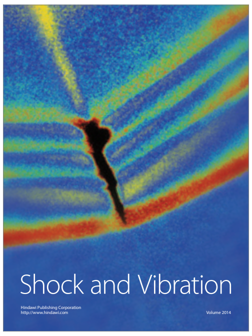

Modelling & Simulation in Engineering

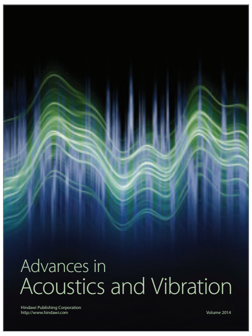
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration