RECOGNITION OF PROJECTED 3D POINTS AND LINES USING A CONTINUOUS GRASP

M. J. HIRSCH, P. M. PARDALOS, AND M. G. C. RESENDE

ABSTRACT. The field of computer vision has experienced rapid growth over the past fifty years. Many computer vision problems have been solved using theory and ideas from algebraic projective geometry. In this research, we look at a previously unsolved problem from object recognition, namely object recognition when the correspondences between the object and image data is not known *a priori*. We formulate this problem as a mixed-integer nonlinear optimization problem in terms of the unknown projection relating the object and image, as well as the unknown assignments of object points and lines to those in the image. The global optimum of this problem recovers the relationship between the object points and lines with those in the image. When certain assumptions are enforced on the allowable projections mapping the object into the image, a proof is provided which permits one to solve the optimization problem via a simple decomposition. We illustrate this decomposition approach on some example scenarios.

1. INTRODUCTION

In the foreword of Hartley and Zisserman [10, p. xi], Faugeras writes ''making a computer see was something that leading experts in the field of artificial intelligence thought to be at the level of difficulty of a summer student's project back in the sixties. Forty years later the task is still unsolved and seems formidable.'' One of the main reasons for this is that the biological vision and recognition process is still largely unknown and therefore hard to emulate on computers. The field of computer vision has grown out of the research directed towards 'making a computer see.' The past 15 years has seen a number of published books and articles presenting a mathematical framework for computer vision, considering problems from the perspective of algebraic projective geometry and invariant theory[4, 5, 9, 10, 15]. In this paper, we consider one of the still unsolved problems in computer vision, namely that of object recognition with unknown correspondence.

The general object recognition problem can be stated as follows: *suppose an object is represented by a set of m_1 points and n_1 lines. Given a picture with m_2 points and n_2 lines, does the picture contain an image of the object, under a projective transformation (i.e., pinhole camera)?* Solutions to this problem have been derived when the correspondence between the object points and lines with those in the picture are known and the scenario is noise-free [8, 9, 10]. However, when the correspondence is not known, and when noise is present in the image data, this problem becomes quite difficult to solve.

There have been some attempts to solve this object recognition problem when certain simplifying assumptions are enforced (e.g., intensity of image points are known and invariant across images [2, 10, 15], errors are not dealt with directly in the problem formulation

Date: January 19, 2007. AT&T Labs Research Technical Report TD-6XLTAY.

Key words and phrases. Object recognition, point/line projection, GRASP, continuous GRASP, global optimization, stochastic local search, nonlinear programming.

[3, 16, 18], the correspondence is already known, and an inlier set is sought to determine the 'best' resulting transformation [2, 10, 15]).

In this paper, we formulate the object recognition problem for points and lines as a mixed-integer nonlinear optimization problem. In particular, we derive equations for point and line correspondence directly accounting for possible noise in the image data. Using a specific type of pinhole camera, we provide a proof by which the optimization problem can be solved via a decomposition technique. This decomposition approach first solves a continuous non-convex global optimization problem to recover the camera parameters (i.e., projection matrix), and then solves a linear assignment problem to determine the correspondence between the 3D points and lines with those in the image.

This paper is set up as follows. In Section 2, we derive equations for the point and line correspondences, explicitly accounting for noise in the image data, and provide a mixed-integer nonlinear optimization formulation for the object recognition problem. Section 3 details the decomposition approach to solve the optimization problem derived in Section 2. In Section 4, we discuss a new heuristic for continuous global optimization. This heuristic is utilized in the first step of the decomposition technique for the scenarios considered in the computational study (Section 5). Concluding remarks are provided in Section 6.

2. Object Recognition Problem Formulation

In this section we begin by deriving equations that will be satisfied for 3-D to 2-D point and line correspondences when error is not present. We then alter these equations to explicitly account for possible uncertainty in the image points and lines, and provide a mixed-integer nonlinear optimization formulation. However, we begin with some notation that will be used throughout this paper.

Let $\alpha = \{\alpha_1, \dots, \alpha_{m_1}, \alpha'_1, \dots, \alpha'_{n_1}\}$ represent a three-dimensional object, i.e., an unordered collection of m_1 points and n_1 lines (without loss of generality, let the unprimed α's represent the points and the primed α's represent the lines). Let $\beta = \{\beta_1, \dots, \beta_{m_2}, \beta'_1, \dots, \beta'_{n_2}\}$ represent a two-dimensional picture, i.e., an unordered collection of m_2 points and n_2 lines (without loss of generality, let the unprimed β's represent the points and the primed β's represent the lines). For convenience, we represent the 3-D object and 2-D image in homogeneous coordinates [8, 9, 10, 17]. Therefore, the 3-D point $\{x, y, z\}$ is represented as the column vector $\{x, y, z, 1\}^T$ and the 2-D point $\{u, v\}$ is represented as the column vector $\{u, v, 1\}^T$. An image formed by a pinhole camera (i.e., projective transformation) can then be represented as a projection from four-dimensional space to three-dimensional space, i.e., as a 3×4 matrix having 9 degrees of freedom [10]. Assume that $\Omega = [\Omega_{ij}]$ is such a 3×4 matrix.

Point Equation Derivation. For an object point $\alpha_i = \{x_i, y_i, z_i, r_i\}^T$ to correspond with the point in the picture $\beta_j = \{u_j, v_j, w_j\}^T$, using Ω as the camera, it must be true that $\Omega\alpha_i = c_{ij}\beta_j$, for some $c_{ij} \neq 0$, where c_{ij} is needed to compensate for the invariance of homogeneous coordinates across scale. If we assume that $w_j \neq 0$ and $\Omega\alpha_i$ does not have its' third coordinate equal to 0 (i.e., $\Omega_{31}x_i + \Omega_{32}y_i + \Omega_{33}z_i + \Omega_{34}r_i \neq 0$), then $\Omega\alpha_i = c_{ij}\beta_j$ is equivalent to the system of Equations (1) - (2). Note that these assumptions on w_j and the third coordinate of $\Omega\alpha_i$ can be enforced in a straight-forward manner [8, 9].

$$\frac{\Omega_{11}x_i+\Omega_{12}y_i+\Omega_{13}z_i+\Omega_{14}r_i}{\Omega_{31}x_i+\Omega_{32}y_i+\Omega_{33}z_i+\Omega_{34}r_i}-\frac{u}{w}=0 \tag{1}$$

$$\frac{\Omega_{21}x_i+\Omega_{22}y_i+\Omega_{23}z_i+\Omega_{24}r_i}{\Omega_{31}x_i+\Omega_{32}y_i+\Omega_{33}z_i+\Omega_{34}r_i}-\frac{u}{w}=0 \tag{2}$$

If $\hat{\Omega\alpha_i}$ denotes the non-homogeneous representation of $\Omega\alpha_i$ and $\hat{\beta}_j$ denotes the non-homogeneous representation of β_j, then the system of Equations (1) - (2) is equivalent to the vector equation $t_{ij} \equiv \hat{\Omega\alpha_i} - \hat{\beta}_j = \mathbf{0}$, which is equivalent to $\|t_{ij}\|_2 = 0$. Hence, when α_i and β_j correspond through the projection Ω, the Euclidean distance between $\hat{\Omega\alpha_i}$ and $\hat{\beta}_j$ will be 0.

Now, suppose that the non-homogeneous image point $\hat{\beta}_j$ has some noise, or uncertainty, associated with it. This uncertainty can be represented as a 2×2 covariance matrix C_j, which we assume is positive semi-definite. Then, even when α_i and β_j correspond through the projection Ω, the Euclidean distance between $\hat{\Omega\alpha_i}$ and $\hat{\beta}_j$ might not be 0. This is due precisely to the uncertainty involved. To account for the uncertainty in the image point, we generalize the Euclidean distance $\|t_{ij}\|_2$ to the Mahalanobis distance $t_{ij}^T C_j^{-1} t_{ij}$ [10]. When this uncertainty is Gaussian in nature (it is generally assumed to be), then we can consider the Gaussian probability density function as representing the likelihood, $F_{ij}(\Omega)$, of α_i and β_j corresponding through the projection Ω.

$$F_{ij}(\Omega) = \frac{1}{2\pi\sqrt{|C_j|}} e^{-\frac{1}{2}t_{ij}^T C_j^{-1} t_{ij}}$$

Line Equation Derivation. Let α'_k be the object line through the two points $P_{k1} = \{x_{k1}, y_{k1}, z_{k1}, r_{k1}\}^T$ and $P_{k2} = \{x_{k2}, y_{k2}, z_{k2}, r_{k2}\}^T$. It is clear that α'_k is defined uniquely by any two distinct points incident to α'_k. Let β'_ℓ be the image line represented as $\beta'_\ell = \{a_\ell, b_\ell, c_\ell\}^T$, where a two-dimensional homogeneous point $q = \{u, v, w\}^T$ is incident to β'_ℓ if and only if $q^T\beta = ua_\ell + vb_\ell + wc_\ell = 0$. Then α'_k and β'_ℓ correspond through the projection Ω precisely when both ΩP_{k1} and ΩP_{k2} are incident to β'_ℓ, i.e., when $(\Omega P_{k1})^T\beta'_\ell = 0$ and $(\Omega P_{k2})^T\beta'_\ell = 0$. Unfortunately, this approach does not generalize to the case when there is error associated with the line β'_ℓ, as these two equations are not metric oriented. Therefore, we will look at the line case from a different perspective.

For $s \in \{1,2\}$, let $\hat{\Omega P_{ks}} = \{u_{ks}, v_{ks}, 1\}^T$ denote the non-homogeneous representation of ΩP_{ks}.

$$u_{ks} = \frac{\Omega_{11}x_{ks}+\Omega_{12}y_{ks}+\Omega_{13}z_{ks}+\Omega_{14}r_{ks}}{\Omega_{31}x_{ks}+\Omega_{32}y_{ks}+\Omega_{33}z_{ks}+\Omega_{34}r_{ks}}$$

$$v_{ks} = \frac{\Omega_{21}x_{ks}+\Omega_{22}y_{ks}+\Omega_{23}z_{ks}+\Omega_{24}r_{ks}}{\Omega_{31}x_{ks}+\Omega_{32}y_{ks}+\Omega_{33}z_{ks}+\Omega_{34}r_{ks}}$$

It can be shown without too much difficulty that the non-homogeneous point incident to β'_ℓ that is closest to $\hat{\Omega P_{ks}}$ is given by

$$\hat{\delta}_{ks\ell} = \{\frac{b_\ell^2 u_{ks} - a_\ell b_\ell v_{ks} - a_\ell c_\ell}{a_\ell^2 + b_\ell^2}, \frac{a_\ell^2 v_{ks} - a_\ell b_\ell u_{ks} - b_\ell c_\ell}{a_\ell^2 + b_\ell^2}\}^T.$$

Defining $t_{ks\ell} = \hat{\delta}_{ks\ell} - \hat{\Omega P_{ks}}$, then we again have, in the error-free case, α'_k and β'_ℓ correspond through the projection Ω precisely when both $t_{k1\ell} = \mathbf{0}$ and $t_{k2\ell} = \mathbf{0}$, i.e., precisely when

$\|t_{k1\ell}\|_2 = \|t_{k2\ell}\|_2 = 0$. Hence, when α'_k and β'_ℓ correspond through the projection Ω, the Euclidean distances from $\hat{\Omega P}_{k1}$ to $\hat{\delta}_{k1\ell}$ and from $\hat{\Omega P}_{k2}$ to $\hat{\delta}_{k2\ell}$ will both be 0.

Now, suppose that each non-homogeneous point on the image line β'_ℓ has some noise, or uncertainty, associated with it. Again, represent this uncertainty by a 2×2 covariance matrix C_ℓ. Therefore, both $\hat{\delta}_{k1\ell}$ and $\hat{\delta}_{k2\ell}$ have an associated covariance matrix C_ℓ. With this uncertainty being Gaussian in nature we can form the likelihood, $\bar{F}_{k\ell}(\Omega)$, of α'_k and β'_ℓ corresponding through the projection Ω.

$$\bar{F}_{k\ell}(\Omega) = \frac{1}{2\pi\sqrt{|C_\ell|}} e^{-\frac{1}{2}(t_{k1\ell}^T C_\ell^{-1} t_{k1\ell} + t_{k2\ell}^T C_\ell^{-1} t_{k2\ell})}$$

Problem Formulation. In addition to the unknown projection matrix, Ω, we also need to determine the correspondence between the 3-D and 2-D points and lines. Hence, for each possible 3-D point α_i and 2-D point β_j, we define a binary variable ψ_{ij} to denote whether α_i and β_j should be in correspondence. We similarly define binary variables $\phi_{k\ell}$ to denote whether the 3-D line α'_k and 2-D line β'_ℓ should be in correspondence. The object recognition problem formulation is then given by Problem (3) - (10), where W is the space of projective transformations. Note that constraints (4), (5), and (8) ensure that each 3D (2D) point will have at most one corresponding 2D (3D) point, and similarly constraints (6), (7), and (9) ensure that each 3D (2D) line will have at most one corresponding 2D (3D) line.

$$\min_{\Omega,\psi,\phi} f(\Omega,\psi,\phi) = -\sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \psi_{ij} F_{ij}(\Omega) - \sum_{k=1}^{n_1}\sum_{\ell=1}^{n_2} \phi_{k\ell}\bar{F}_{k\ell}(\Omega) \tag{3}$$

$$\ni \sum_{i=1}^{m_1} \psi_{ij} \le 1 \;\forall\; j \in \{1,\ldots,m_2\} \tag{4}$$

$$\sum_{j=1}^{m_2} \psi_{ij} \le 1 \;\forall\; i \in \{1,\ldots,m_1\} \tag{5}$$

$$\sum_{k=1}^{n_1} \phi_{k\ell} \le 1 \;\forall\; \ell \in \{1,\ldots,n_2\} \tag{6}$$

$$\sum_{\ell=1}^{n_2} \phi_{k\ell} \le 1 \;\forall\; k \in \{1,\ldots,n_1\} \tag{7}$$

$$\psi_{ij} \in \{0,1\} \;\forall\; i \in \{1,\ldots,m_1\}, \forall\; j \in \{1,\ldots,m_2\} \tag{8}$$

$$\phi_{k\ell} \in \{0,1\} \;\forall\; k \in \{1,\ldots,n_1\}, \forall\; \ell \in \{1,\ldots,n_2\} \tag{9}$$

$$\Omega \in W \tag{10}$$

3. Problem Decomposition

A general pinhole camera, Ω, can be represented by $\Omega = K[R|t]$, where K is an internal camera calibration matrix (of size 3×3, with 3 degrees of freedom), R is a 3-D rotation matrix (of size 3×3, with 3 degrees of freedom), and t is a translation vector (of size 3×1, with 3 degrees of freedom) [10]. When K is set to the identity matrix, then the resulting projection matrix Ω preserves distances between points. In what follows, we make use of the following theorem.

Theorem 3.1. *Let $\tilde{f}(\Omega) = -\sum_{i=1}^{m_1}\sum_{j=1}^{m_2} F_{ij}(\Omega) - \sum_{k=1}^{n_1}\sum_{\ell=1}^{n_2} \bar{F}_{k\ell}(\Omega)$ and N_c^p (N_c^q) represent the truthful number of point (line) correspondences between the object and image. Suppose* *the following:* ***i)*** *Ω preserves distances between points, i.e., $\|x-y\|_2 = \|\Omega(x)-\Omega(y)\|_2$,* ***ii)*** *$N_c = N_c^p + N_c^q \geq dof(\Omega)$, and* ***iii)*** *The object point and line geometry is sufficiently random.*

Then, in the case where there is no noise in the image (i.e., covariance matrices are **0***), the Ω that minimizes $\tilde{f}(\Omega)$ is precisely the Ω mapping the points and lines of the 3D object onto their truthful correspondences in the image.*

Proof. Let $\tilde{f}'(\Omega)$ equal $\tilde{f}(\Omega)$ when no random noise is present in the image. Then

$$\begin{aligned}\tilde{f}'(\Omega) &= \lim_{\substack{C_j\to 0\,\forall\, j\\ C_\ell\to 0\,\forall\,\ell}} \tilde{f}(\Omega)\\ &= \lim_{\substack{C_j\to 0\,\forall\, j\\ C_\ell\to 0\,\forall\,\ell}} \Big[-\sum_{i=1}^{m_1}\sum_{j=1}^{m_2} F_{ij}(\Omega) - \sum_{k=1}^{n_1}\sum_{\ell=1}^{n_2} \bar{F}_{k\ell}(\Omega)\Big]\\ &= -\sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \lim_{C_j\to 0\,\forall\, j} F_{ij}(\Omega) - \sum_{k=1}^{n_1}\sum_{\ell=1}^{n_2} \lim_{C_\ell\to 0\,\forall\,\ell} \bar{F}_{k\ell}(\Omega)\\ &= -\sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \delta(\|t_{ij}\|^2) - \sum_{k=1}^{n_1}\sum_{\ell=1}^{n_2} \delta(\|t_{k\ell}\|^2)\end{aligned}$$

where $\delta(x)$ is the Dirac-delta, or unit-impulse, function, i.e., $\delta(x-a)=0$ if $x\neq a$ and $\int_{-\infty}^{\infty}\delta(x)\mathtt{dx}=1$.

Now, $N_c = N_c^p + N_c^q \leq \min[m_1,m_2] + \min[n_1,n_2]$. Without loss of generality, assume that the first N_c^p object points correspond with the first N_c^p image points and the first N_c^q object lines correspond with the first N_c^q image lines.

For each $i\in\{1,\ldots,m_1\},\ j\in\{1,\ldots,m_2\}$, let $\Omega_{ij} = \{\Omega\in W \mid \hat{\beta}_j = \widehat{\Omega\alpha_i}\}$. Similarly, for each $k\in\{1,\ldots,n_1\},\ \ell\in\{1,\ldots,n_2\}$, let $\check{\Omega}_{k\ell} = \{\Omega\in W \mid (\Omega P_{k1})^T\beta'_\ell = 0,\ (\Omega P_{k2})^T\beta'_\ell = 0,$ P_{k1} and P_{k2} distinct points incident to $\alpha'_k\}$. Since there is no random noise and $N_c \geq dof(\Omega)$, $\bar{\Omega} = \Big[\bigcap_{i\in\{1,\ldots,N_c^p\}}\Omega_{ii}\Big]\bigcap\Big[\bigcap_{k\in\{1,\ldots,N_c^q\}}\check{\Omega}_{kk}\Big]$ is well-defined and unique.

Choose any sets of pairs $\{\{i_1,j_1\},\ldots,\{i_r,j_r\}\}$ and $\{\{k_1,\ell_1\},\ldots,\{k_s,\ell_s\}\}$ with the following properties:
a) $i_t\in\{1,\ldots,m_1\}\ \forall\, t\in\{1,\ldots,r\}$ and $i_1,\ldots,i_r$ are distinct;
b) $j_t\in\{1,\ldots,m_2\}\ \forall\, t\in\{1,\ldots,r\}$ and $j_1,\ldots,j_r$ are distinct;
c) $k_t\in\{1,\ldots,n_1\}\ \forall\, t\in\{1,\ldots,s\}$ and $k_1,\ldots,k_s$ are distinct;
d) $\ell_t\in\{1,\ldots,n_2\}\ \forall\, t\in\{1,\ldots,s\}$ and $\ell_1,\ldots,\ell_s$ are distinct;
e) $r+s\geq dof(\Omega)$;
f) $\Big(\exists\, t\in\{1,\ldots,r\}\ni i_t\neq j_t\Big)\vee\Big(\exists\, t\in\{1,\ldots,s\}\ni k_t\neq \ell_t\Big)$;

Since the 3D point and line geometry is sufficiently random,

$$\mathbf{P}\Big(\Big[\bigcap_{t\in\{1,\ldots,r\}}\Omega_{i_t j_t}\Big]\bigcap\Big[\bigcap_{t\in\{1,\ldots,s\}}\check{\Omega}_{k_t\ell_t}\Big] = \emptyset\Big) = 1.$$

Therefore, for any sets of pairs $\{\{i_1,j_1\},\ldots,\{i_r,j_r\}\}$ and $\{\{k_1,\ell_1\},\ldots,\{k_s,\ell_s\}\}$ satisfying the above conditions a) through f), the probability that there exists an Ω projecting each of the 3D points α_{i_t} onto the 2D points β_{j_t} $(t=1,\ldots,r)$ and each of the 3D lines α'_{k_t} onto the 2D lines β'_{ℓ_t} $(t=1,\ldots,s)$ is zero. Hence, the global minimum of $\tilde{f}'(\Omega)$ occurs at $\Omega=\bar{\Omega}$, i.e., at the Ω mapping the N_c^p 3D points and N_c^q 3D lines onto their truthful correspondences. $\square$

Theorem 3.1 provides the basis of our decomposition approach. When there is no noise present in the image we can solve the optimization Problem (3) - (10) by first finding the Ω that minimizes $\tilde{f}(\Omega)$. Projecting all the 3D points and lines into the image via the recovered Ω, the second step is to apply a linear assignment algorithm to determine the 3D to 2D point and line correspondences. We make note of the following. First, $\tilde{f}(\Omega)$ is non-convex in Ω, and while there exist algorithms for finding a *local minimum* in a finite number of iterations, it is well known that finding the global minimum is inherently unsolvable in a finite number of steps [1, 19]. Second, in practice, there is normally some noise in the image data, thus making the global minimum of $\tilde{f}(\Omega)$, only an approximation to the true projection matrix. To solve the first step in the decomposition approach, we make use of a new global optimization heuristic, Continuous GRASP, as detailed in Section 4. The linear assignment algorithm used for the second step of the decomposition is an implementation of the JVC algorithm [14].

In summary, the decomposition approach we propose to solve Problem (3) - (10) is given by:

(1) Use the C-GRASP heuristic to find the Ω that minimizes $\tilde{f}(\Omega)$.
(2) Apply the JVC linear assignment algorithm to determine the association between the 3D points and lines with those in the image.

4. Continuous GRASP

Feo and Resende [6, 7] describe the metaheuristic GRASP (greedy random adaptive search procedures) as a multi-start local search procedure, where each GRASP iteration consists of two phases, a construction phase and a local search phase. In the construction phase, interactions between greediness and randomization generate a diverse set of good-quality solutions. The local search phase attempts to improve the solutions found by construction. The best solution over all of the multi-start iterations is retained as the final solution.

Hirsch et al. [11] describe Continuous GRASP (C-GRASP), an adaptation of GRASP to solve continuous global optimization problems. C-GRASP works by discretizing the domain into a uniform grid. Both the construction and local improvement phases move along points on the grid. As the algorithm progresses, the grid adaptively becomes more dense. C-GRASP resembles GRASP in that it is a multi-start stochastic search metaheuristic that uses a randomized greedy construction procedure to generate starting solutions for a local improvement algorithm. The main difference is that an iteration of C-GRASP does not consist of a single greedy randomized construction followed by local improvement, but rather a series of construction-local improvement cycles with the output of construction serving as the input of the local improvement, as in GRASP. Unlike GRASP, however, the output of the C-GRASP local improvement procedure serves as the input of the C-GRASP construction procedure.

Hirsch et al. [13] proposed modifications to the original C-GRASP algorithm, resulting in a significant decrease in the number of objective function evaluations required to converge to the global optimum. The modified C-GRASP heuristic was applied to a sensor registration problem in [12], which is similar to the object recognition problem considered presently. In the remainder of this section, we detail this version of the C-GRASP heuristic.

4.1. **The heuristic.** Pseudo-code for the C-GRASP heuristic is shown in Figure 1. The procedure takes as input the problem dimension n, lower and upper bound vectors ℓ and u, the objective function $f(\cdot)$, as well as the parameters `MaxRS`, h_s, h_e, and ρ_{lo}. `MaxRS`

```
procedure C-GRASP(n, ℓ, u, f(·), MaxRS, h_s, h_e, ρ_lo)
1    f* ← ∞;
2    for j = 1, ..., MaxRS do
3        x ← UnifRand(ℓ, u);
4        h ← h_s;
5        while h ≥ h_e do
6            Impr_C ← false;
7            Impr_L ← false;
8            [x, Impr_C] ← ConstructGreedyRandomized(x, f(·), n, h, ℓ, u, Impr_C);
9            [x, Impr_L] ← LocalImprovement(x, f(·), n, h, ℓ, u, ρ_lo, Impr_L);
10           if f(x) < f* then
11               x* ← x;
12               f* ← f(x);
13           end if
14           if Impr_C = false and Impr_L = false then
15               h ← h/2;    /* make grid more dense */
16           end if
17       end while
18   end for
19   return(x*);
end C-GRASP;
```

FIGURE 1. Pseudo-code for C-GRASP.

defines the number of multi-starts to perform in the C-GRASP algorithm, h_s and h_e define the starting and ending discretization levels for each multi-start, and ρ_{lo} defines the portion of the neighborhood of the current solution that is searched during the local improvement phase of the algorithm.

Line 1 of the pseudo-code initializes the objective function value f^* of the best solution found to infinity. We perform `MaxRS` independent multi-starts. For each multi-start, Line 3 initializes the solution x to a random point distributed uniformly over the box in $\mathbb{R}^n$ defined by ℓ and u. The parameter h, that controls the discretization density of the search space, is initialized to h_e in line 4. The code sequentially executes lines 6 to 16 as long as $h \geq h_e$. The construction and local improvement phases are then called in lines 8 and 9, respectively. The solution returned from the local improvement procedure is compared against the current best solution in line 10. If the returned solution has a smaller objective value than the current best solution, then, in lines 11– 12, the current best solution is updated with the returned solution. In line 14, if the variables $\texttt{Impr}_\texttt{C}$ and $\texttt{Impr}_\texttt{L}$ are still set to `false`, then the grid density is increased by halving h, in line 15. The variable $\texttt{Impr}_\texttt{C}$ is `false` upon return from the construction procedure if and only if no improvement is made in the construction phase. Section 4.3 shows that the $\texttt{Impr}_\texttt{L}$ variable is `false` on return from the local improvement procedure if and only if the input solution to local improvement is determined to be an *h-local minimum*. We increase the grid density at this stage because repeating the construction procedure with the same grid density will not improve the solution. This approach allows C-GRASP to start with a coarse discretization and adaptively increase the density as needed, thereby intensifying the search in a more dense discretization when a good solution has been found. The best solution found, at the end of the `MaxRS` multi-starts, is returned in line 19.

4.2. **Construction procedure.** In this section, we describe in detail the construction procedure. The construction algorithm combines greediness and randomization to produce a diverse set of good-quality solutions from which to start the local improvement phase. The construction algorithm is shown in Figure 2. The input is a solution vector x. To start, line 1 of the algorithm allows all coordinates of x to change in the current construction call (i.e. these coordinates are unfixed). In turn, in line 10 of the pseudo-code, if `ReUse` is `false`, a line search is performed in each unfixed coordinate direction i of x with the other $n-1$ coordinates of x held at their current values. In lines 10 and 11 of the pseudo-code, the value z_i for the i-th coordinate that minimizes the objective function, together with the objective function value g_i, are saved. In line 11, $\check{x}^i$ denotes x with the i-th coordinate set to z_i.

After looping through all unfixed coordinates (lines 7–16), in lines 17–23 a restricted candidate list (RCL) is formed containing the unfixed coordinates i whose g_i values are less than or equal to $\underline{g}+\alpha\cdot(\bar{g}-\underline{g})$, where $\bar{g}$ and $\underline{g}$ are, respectively, the maximum and minimum g_i values over all currently unfixed coordinates of x, and $\alpha\in[0,1]$ is randomly determined in line 2. In line 24, a coordinate is chosen at random from the `RCL`, say coordinate $j\in$ `RCL`. Line 25 checks whether x_j and z_j are equal. If so, line 26 sets `ReUse` to the value `true`. Otherwise, in lines 28–30, `ReUse` is set to `false`, $\texttt{Impr}_\texttt{C}$ is set to `true`, and x_j is set to equal z_j. Finally, in line 30, we fix coordinate j of x, by removing j from the set `UnFixed`. Choosing a coordinate by selecting at random from the `RCL` ensures both greediness and randomness in the construction phase. The above procedure is continued until all of the n coordinates of x have been fixed. At that stage, x and $\texttt{Impr}_\texttt{C}$ are returned from the construction procedure.

Note that the `ReUse` variable is utilized to speed up computations by avoiding unnecessary line searches. More details can be found in Hirsch et al [13]. The parameter α controls the size of the `RCL` and therefore determines the mix of greediness and randomness in the construction procedure. Different values of α throughout the run allow some construction phases to be more greedy while others to be more random.

4.3. **Local improvement procedure.** C-GRASP makes no use of derivatives. Though derivatives can be easily computed for many functions, they are not always available or efficiently computable for all functions. The local improvement phase (with pseudo-code shown in Figure 3) can be seen as *approximating* the role of the gradient of the objective function $f(\cdot)$. From a given input point $x\in\mathbb{R}^n$, the local improvement algorithm generates a neighborhood, and determines at which points in the neighborhood, if any, the objective function improves. If an improving point is found, it is made the current point and the local search continues from the new solution.

Let $\bar{x}\in\mathbb{R}^n$ be the current solution and h be the current grid discretization parameter. Define

$$S_h(\bar{x}) = \{x\in S \mid \ell\le x\le u,\ x=\bar{x}+\tau\cdot h,\ \tau\in\mathbb{Z}^n\}$$

to be the set of points in S that are integer steps (of size h) away from $\bar{x}$. Let

$$B_h(\bar{x}) = \{x\in S \mid x=\bar{x}+h\cdot(x'-\bar{x})/\|x'-\bar{x}\|,\ x'\in S_h(\bar{x})\setminus\{\bar{x}\}\}$$

be the projection of the points in $S_h(\bar{x})\setminus\{\bar{x}\}$ onto the hyper-sphere centered at $\bar{x}$ of radius h. The *h-neighborhood* of the point $\bar{x}$ is defined as the set of points in $B_h(\bar{x})$.

```
procedure ConstructGreedyRandomized(x, f(·), n, h, ℓ, u, Impr_C)
    UnFixed ← {1, ..., n};
    α ← UnifRand(0, 1);
    ReUse ← false;
    while UnFixed ≠ ∅ do
        g_ ← +∞;
        ḡ ← −∞;
        for i = 1, ..., n do
            if i ∈ UnFixed then
                if ReUse = false then
                    z_i ← LineSearch(x, h, i, n, f(·), ℓ, u);
                    g_i ← f(x̌^i);
                end if
                if g_ > g_i then g_ ← g_i;
                if ḡ < g_i then ḡ ← g_i;
            end if
        end for
        RCL ← ∅;
        Threshold ← g_ + α · (ḡ − g_);
        for i = 1, ..., n do
            if i ∈ UnFixed and g_i ≤ Threshold then
                RCL ← RCL ∪ {i};
            end if
        end for
        j ← RandomlySelectElement(RCL);
        if x_j = z_j then
            ReUse ← true;
        else
            x_j ← z_j;
            ReUse ← false;
            Impr_C ← true;
        end if
        UnFixed ← UnFixed \ {j};      /* Fix coordinate j. */
    end while
    return(x, Impr_C);
end ConstructGreedyRandomized;
```

FIGURE 2. Pseudo-code for C-GRASP construction phase.

The local improvement procedure is given a starting solution $x \in S \subseteq \mathbb{R}^n$. The current best local improvement solution x^* is initialized to x in line 1. Lines 3 and 4 determine the number of grid points, based on the current value of the discretization parameter h, and the maximum number of points in $B_h(x^*)$ that are to be examined. This number of grid points is defined by the parameter ρ_{lo} which is the portion of the neighborhood which is to be examined. If all of these points are examined and no improving point is found, the current solution x^* is considered an *h-local minimum*.

procedure `LocalImprovement`$(x, f(\cdot), n, h, \ell, u, \rho_{lo}, \mathtt{Impr_L})$
1 $x^* \leftarrow x$;
2 $f^* \leftarrow f(x)$;
3 `NumGridPoints` $\leftarrow \prod_{i=1}^{n} \lceil (u_i - \ell_i)/h \rceil$;
4 `MaxPointsToExamine` $\leftarrow \lceil \rho_{lo} \cdot$ `NumGridPoints`$\rceil$;
5 `NumPointsExamined` $\leftarrow 0$;
6 **while** `NumPointsExamined` $\leq$ `MaxPointsToExamine` **do**
7 `NumPointsExamined` $\leftarrow$ `NumPointsExamined` $+ 1$;
8 $x \leftarrow$ `RandomlySelectElement`$(B_h(x^*))$;
9 **if** $\ell \leq x \leq u$ **and** $f(x) < f^*$ **then**
10 $x^* \leftarrow x$;
11 $f^* \leftarrow f(x)$;
12 $\mathtt{Impr_L} \leftarrow$ `true`;
13 `NumPointsExamined` $\leftarrow 0$;
14 **end if**
15 **end while**
16 **return**$(x^*, \mathtt{Impr_L})$;
end `LocalImprovement`;

FIGURE 3. Pseudo-code for C-GRASP local improvement phase.

TABLE 1. Object recognition scenario classes.

Scenario	Points				Lines			
Class	3D	2D	True Corr.	Max Noise	3D	2D	True Corr.	Max Noise
1	15	20	11	0.5	-	-	-	-
2	12	16	8	0.5	8	11	5	0.5
3	-	-	-	-	16	25	13	0.5

Starting at the point x^*, in the loop in lines 6–15, the algorithm randomly selects `MaxPointsToExamine` points in $B_h(x^*)$, one at a time. In line 9, if the current point x selected from $B_h(x^*)$ is feasible and is better than x^*, then x^* is set to x, $\mathtt{Impr_L}$ is set to `true`, and the process restarts with x^* as the starting solution. $\mathtt{Impr_L}$ is used to determine whether the local improvement procedure improved the best solution. Local improvement is terminated when an *h-local minimum* solution x^* is found. At that point, x^* and $\mathtt{Impr_L}$ are returned from the local improvement procedure.

5. Computational Study

To illustrate our approach for solving the object recognition optimization problem 3 - 10, we consider three scenarios classes. Each class is designated by the number of 3D and 2D points and lines, the number of truthful 3D to 2D point and line correspondences, and the maximum noise (in pixels) associated with the 2-D data. The details for each scenario class are given in Table 1.

For each class, 100 randomly generated scenario instances were created. Each randomly generated scenario instance was input to the C-GRASP algorithm and run for a fixed number of multi-starts. The C-GRASP parameters are listed in Table 2. When the C-GRASP algorithm completed the run for a scenario instance, using the best projection matrix found, Ω, the 3-D points and lines from the scenario were projected onto the 2-D

TABLE 2. Object recognition C-GRASP parameters.

Number Multi-Starts	100
h_s	0.10
h_e	0.01
ρ_{lo}	0.70

TABLE 3. Object recognition test results.

Scenario				Points		Lines	
Class	ATT	ABT	ABI	ANCC	ADC	ANCA	ADC
1	9.11	4.97	54.7	9.1	1.12	-	-
2	13.17	8.31	63.6	7.2	0.81	4.5	0.05
3	30.58	18.28	57.6	-	-	8.4	0.76

image plane, and the JVC linear assignment algorithm [14] was applied to determine the optimal assignment of 3-D points and lines to those in 2-D.

For each scenario class, we averaged (over all the scenario instances in the class) the time needed (in minutes) for the C-GRASP algorithm to complete all multi-starts (ATT), the time (in minutes) when C-GRASP found the best Ω (ABT), the multi-start iteration when C-GRASP found the best Ω (ABI), the number of assignments of 3-D points and lines to 2-D points and lines that were correct (ANCA), and the distances between the projected 3-D points and lines and their truthful corresponding 2-D points and lines (ADC). These results are shown in Table 3. As is clear from the table, for all three scenario classes, the decomposition approach performs very well. Of note is that adding lines to the scenario seems to add significant complexity to the surface of the objective function, thus taking longer for the C-GRASP algorithm to run to completion and longer on average to locate the correct solution, from a time and multi-start perspective.

6. CONCLUSIONS

In this paper, we have examined a problem from computer vision: the recognition of a 3-D object, represented by points and lines in an image, when the correspondence of points and lines is not known *a priori*. We have formulated this problem as a mixed-integer nonlinear optimization problem, explicitly accounting for possible noise in the 2-D data. For a pinhole camera (with specific internal calibration parameters), we decomposed the mixed-integer nonlinear optimization problem into a two step process of first determining the best projection matrix transforming the 3-D points and lines onto the 2D picture and then using a linear assignment algorithm to determine the correspondences between the points and lines of the object and those of the image. Computational studies have shown that this approach does a very good job of determining the correct projection matrix and correspondences. Future research will be geared towards reducing the time of the C-GRASP heuristic, in an effort to make this approach suitable for a real-time system.

REFERENCES

[1] C. G. E. Boender and H. E. Romeijn. Stochastic methods. In R. Horst and P. M. Pardalos, editors, *Handbook of Global Optimization*, pages 829–869. Kluwer Academic Publishers, 1995.

[2] J. Chai and S. D. Ma. Robust epipolar geometry estimation using genetic algorithm. *Pattern Recognition Letters*, 19:829–838, 1998.

[3] Y. Cheng, V. Wu, R. T. Collins, A. R. Hanson, and E. M. Riseman. Maximum weight bipartite matching technique and its application to solve image feature matching. In *Proc. of the SPIE Conference on Visual Communication and Image Processing*, 1996.

[4] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press, 1993.

[5] O. Faugeras and Q. T. Luong. *The Geometry of Multiple Images: The laws that govern the formation of multiple images of a scene and some of their applications*. The MIT Press, 2001.

[6] T. A. Feo and M. G. C. Resende. A probabilistic heuristic for a computationally difficult set covering problem. *Operations Research Letters*, 8:67–71, 1989.

[7] T. A. Feo and M. G. C. Resende. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6:109–133, 1995.

[8] R. F. Gleeson, F. D. Grosshans, M. J. Hirsch, and R. M. Williams. Algorithms for the recognition of 2D images of m points and n lines in 3D. *Image and Vision Computing*, 21:497–504, 2003.

[9] F. D. Grosshans. On the equations relating a three-dimensional object and its two-dimensional images. *Advances in Applied Mathematics*, 34:366–392, 2005.

[10] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[11] M. J. Hirsch, C. N. Meneses, P. M. Pardalos, and M. G. C. Resende. Global optimization by Continuous GRASP. *To appear in Optimization Letters*, 2006.

[12] M. J. Hirsch, P. M. Pardalos, and M. G. C. Resende. Sensor registration in a sensor network by Continuous GRASP. In *Proc. of the IEEE Military Communications Conference (MILCOM)*, 2006.

[13] M. J. Hirsch, P. M. Pardalos, and M. G. C. Resende. Speeding up Continuous GRASP. *Submitted to European Journal of Operational Research*, 2006.

[14] R. Jonker and A. Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 39:325–340, 1987.

[15] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. Springer-Verlag, 2004.

[16] G. L. Scott and H. C. Longuet-Higgins. An algorithm for associating the features of two images. *Proc. of the Royal Society of London **B***, 244:21–26, 1991.

[17] J. G. Semple and G. T. Kneebone. *Algebraic Projective Geometry*. Oxford University Press, 1952.

[18] L. S. Shapiro and J. M. Brady. Feature based correspondence: an eigenvector approach. *Image and Vision Computing*, 10(5):283–288, 1992.

[19] L. Zhan. Fast stochastic global optimization methods and their applications to cluster crystallization and protein docking. *PhD. thesis, University of Waterloo*, 2005.

(Michael J. Hirsch) RAYTHEON, INC., NETWORK CENTRIC SYSTEMS, P.O. BOX 12248, ST. PETERSBURG, FL, 33733-2248, AND DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING, UNIVERSITY OF FLORIDA, 303 WEIL HALL, GAINESVILLE, FL, 32611, USA.

E-mail address: mjh8787@ufl.edu

(Panos M. Pardalos) DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING, UNIVERSITY OF FLORIDA, 303 WEIL HALL, GAINESVILLE, FL, 32611, USA.

E-mail address: pardalos@ufl.edu

(Mauricio G. C. Resende) ALGORITHMS AND OPTIMIZATION RESEARCH DEPARTMENT, AT&T LABS RESEARCH, 180 PARK AVENUE, ROOM C241, FLORHAM PARK, NJ 07932 USA.

E-mail address: mgcr@research.att.com