POSE DETERMINATION FROM A SINGLE IMAGE IN A CONTROLLED CAD ENVIRONMENT

A. D. Styliadis
The ATEI of Thessaloniki
Dept. of Informatics
Agiou Panteleimona 24
55133, Thessaloniki, GR
styl@it.teithe.gr

C. J. Karagiannidis
The ATEI of Thessaloniki
Dept. of Informatics
Agiou Panteleimona 24
55133, Thessaloniki, GR
karagian@acm.org

N. C. Zestas
The Aristotle University of Thessaloniki
P.O. Box 469
54124, Thessaloniki, GR
nzestas@topo.auth.gr

K. N. Chatzara
Synthesis Interactive
Katouni 28
554 25, Thessaloniki, Greece
synthe@otenet.gr

ABSTRACT

The purpose of this work is to derive an easy-to-use interactive method, using simple CAD software, for sensor attitude (*camera pose determination*, i.e. camera positioning (*Xo*, *Yo*, *Zo*) and orientation (ω, φ, κ rotations)) and object pose determination, with a 10^{-3} accuracy, from a single image. It is actually a control-point-free method intended for terrestrial photogrammetric engineering applications (mainly in Architecture and Archaeology dealing with facades) and for robotics applications (mainly in robot location-tracking and object recognition).

This new method is based on mutually parallel or perpendicular line pairs in observed rectangular shape images usually found in photography. In man-made environments, rectangular shapes can be seen everywhere. It is thus convenient to use rectangular shapes for pose and object determination in photogrammetric engineering (close-range space rejection) and robotics (robot location).

The proposed system is to some degree automatic, since it analytically computes discrepancy vectors and examines possible solutions to minimize a penalty function. These different solutions are animated on the screen and the procedure is simply stopped by the user when he feels the solution was achieved. When the solution deviates user interaction is possible. As a supplement to automatic procedure a manual approach is provided for micro-corrections and enhancements. This manual approach is based on mouse movements and appropriate push-button selections according to a predefined step.

Some merits of the proposed method are: Single photography of known rectangular shapes is sufficient. Multiple rectangular shapes can be utilized to promote the accuracy. Partial rectangular shapes, usually found in old photography, also can be utilized. Finally, the major merit of the method is that the solution can be uniquely determined in an interactive user-friendly CAD environment.

Experimental results show the feasibility of the proposed approach. Accuracy evaluation for determining pose determination precision is also included.

Keywords: CAD programming, image understanding, camera calibration, pose determination, virtual camera.

1. INTRODUCTION

The computation of the *position* and *orientation* of an object (well known in CAD and robotics as the *object pose* problem) has important applications, such as camera calibration, sensor attitude determination (in digital photogrammetry), tracking and object recognition (in robotics), etc. This computation is based on images of feature points in photography when the geometric configuration of the object is known in advance. Researchers have coined the term *Perspective-n-Point problem* (or P*n*P problem) for this type of problem with *n* feature points. Recent approaches have formulated iterative and closed-form object pose solutions when a few feature points are considered in coplanar and/or noncoplanar configurations [DeM95], [Abid96], [Styl99].

Journal of WSCG, Vol.11, No. , ISSN 1213-6972

WSCG'2003, February 3-7, 2003, Plzen, Czech Republic.

However, <u>interactive</u> virtual CAD camera-based pose computations, which make use of rectangular shapes of known geometric attributes, found in a single image (photography), may be more *practical* because the closed-form solutions are based on more than one images, and more *robust* because the iterative solutions rely on the Newton-Raphson method. This method presents two significant drawbacks: first, an approximate pose must be provided to initiate the iteration process and second, at each iteration step, the pseudoinverse matrix of a Jacobian of dimensions 2N*6 (where N is the number of feature points) must be found; a computationally expensive operation.

In 3-D space an object has six degrees of freedom, which correspond to one rotation and one translation around and along each axis of the reference coordinate system. To determine the values of these six parameters, it is necessary to consider the matchings between at least three image lines and three model lines. Recently, many people have worked on this problem and generally they tried to interpret the *three image line junctions* as projections of trihedral vertices of the model. Researchers have solved this problem in an analytical way in the case of orthographic projection. But the correct model for human and camera vision is perspective projection. This is more constraining since it permits us to suppress the ambiguity blinded with the orthographic projection known as the “Necker’s cube illusion.”

In digital *terrestrial* photogrammetric applications (unlike the *aerial* case where the sensor rotations are normally small) due to the complexity of the cases, the camera rotations can range to any possible values. This creates a major problem in defining good initial values for the ω, φ, κ angles to be entered to the subsequent photogrammetric adjustment, initially for *camera pose determination* and, after that, for the *object pose determination* using the close-range space resection closed-form algorithms.

The problem is even worse in Architectural and Archaeological applications, where normally the use of low-end equipment is desirable or simply used. Under these circumstances the number of *control points* is relaxed, the cameras are of many types (from amateurs to camcorders), surveying instruments may not be used, and generally control information is reduced to simple distance measurements for scale determination. Thus ω, φ, κ rotations are rarely recorded and the determination of their initial values becomes an empirical, time-consuming and often very difficult job, many times performed by people (like architects and archaeologists) with little photogrammetry experience [Alv02], [Smin02].

Nowadays, CAD vendors and current software provide the possibilities of entering raster images as well as easy manipulation of 3-D vector graphics. Furthermore, in CAD environments, friendly user interfaces using pop-up windows and dialog boxes give the user the ability to view photography and 3-D design files - in *top*, *front*, *isometric* and *side* views - at the same time. The proposed system profits from these advances taking ideas and utilities in an attempt to solve a classical photogrammetric engineering and robotics problem, such as the determination of sensor attitude without control points, using CAD tools and event-driven programming techniques. In photogrammetric engineering and robotics the calculation of approximated values for exterior orientation parameters is a difficult and time-consuming task. The development system is based on a simple commercial CAD environment, enhanced by dialog boxes and hook functions implemented in an event-driven manner using the C++ programming language. The main C++ enhancements have been gathered in an easy-to-use dialog box (see Fig. 4) offering a friendly GUI support [Styl02].

The whole operational procedure is based on <u>two facade's models attempting to match each other on screen</u> (CAD environment – design session). These models are: the *red TARGET model*, which is built according to provided image co-ordinates, and the *generic CAD model*, which is built according to any available distance measurements for the object (façade). This matching is achieved after a continuous stepwise CAD digital (virtual) camera's positioning and orientation following the traditional analytical photogrammetric engineering technique. This CAD camera is actually a virtual camera provided by the CAD platform and controlled by C software in order to achieve exterior orientation (i.e. *camera pose determination*) by giving minimum overall discrepancies for these two models.

The proposed method makes use of rectangular shapes, which are very common in Architecture and Archaeology (known as *facades*). Examples of objects containing rectangular shapes include: windows, doors, walls, corridors, etc. Some merits of the proposed approach are as follows:

- single photography of known (geometrically) rectangular shapes is sufficient for good initial values for pose determination
- it is a control-point free method
- partial rectangular shapes, found usually in old photography, can also be utilized, as long as only two points and three line segments are available
- it is an interactive user-oriented method
- in an available photography, the corner points and the lines through the corners can be detected automatically using digital image processing techniques
- no complicated algebra/trigonometry is involved; this saves a lot of computation time.

The inaccuracy of the proposed CAD-based rectification is about 0.09 % (i.e. 10^{-3} accuracy). This inaccuracy was determined empirically by comparing objects' edge-distances, calculated using the eight rectification parameters produced by the proposed system (space resection parameters), with the similar edge-distances calculated by the classical analytical photogrammetric method (known as *collinearity method* in photogrammetry) (see Table 1).

2. PROBLEM DEFINITION

2.1 The Viewer Centered Coordinate System

All the co-ordinate systems used throughout this paper are left-handed systems. Among them, the viewer centered coordinate system is defined with the *camera lens center* or the *observer* located at the origin and such that the view axis is collinear to the z axis. To keep the image in a positive orientation, it is assumed that the image plane is perpendicular to the view axis, in front of the viewer and at a distance f equal to the *focal length* (see Fig.1). Using the perspective model, the image of any point in space is equal to the intersection of the image plane and the line joining the point to the center of the camera lens. Thus if Pi is a point of coordinates (X_i, Y_i, Z_i), its image corresponds to the point pi of coordinates (x_i, y_i, z_i) = ($X_i * f / Z_i$, $Y_i * f / Z_i$, f). By convention, the capital and the small letters will refer, respectively, to *space* and *image* elements.

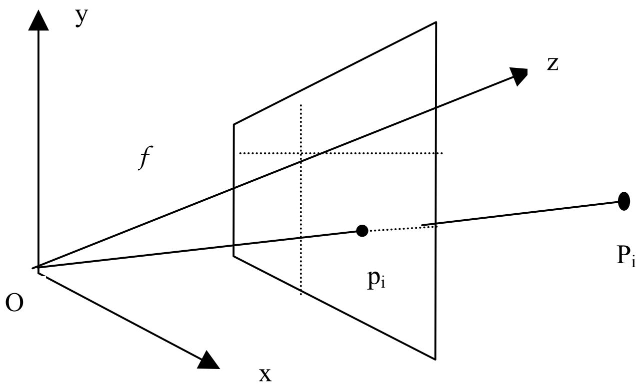

Figure 1. Viewer Centered Coordinate System.

2.2 *The Inverse Perspective Projection of an Image Line*

Let us consider an image line li characterized by a vector ui of components (ai, bi, 0) and a point pi of coordinates (xi, yi, f). This line corresponds to the projection of a space line Li. The perspective projection model constrains line Li to lie in a particular plane passing through the origin of the coordinate system and containing the image line li (see Fig. 2). This plane is called the "interpretation plane". A vector Ni normal to this plane can be computed. It is equal to the product of vector v_i and the vector defined by the origin and point p_i:

$$\begin{bmatrix} a_i \\ b_i \\ 0 \end{bmatrix} \times \begin{bmatrix} x_i \\ y_i \\ f \end{bmatrix} = \begin{bmatrix} b_i f \\ -a_i f \\ a_i y_i - b_i x_i \end{bmatrix}$$

It is interesting to note that the third coordinate of the vector is equal to the Euclidean distance d_i between the center of the image and line l_i. Let the vector V_i of components (A_i, B_i, C_i) be the direction vector of line L_i. Its components must verify the following equation (eq. 1) corresponding to the orthogonality of N_i and V_i [Dhom89].

$$A_i * b_i - B_i * a_i + C_i * d_i / f = 0 \quad \text{(eq. 1)}$$

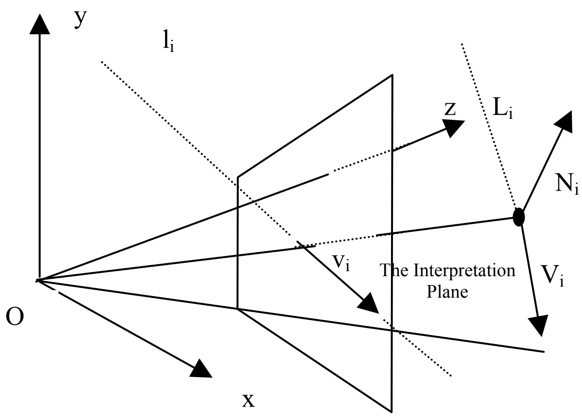

Figure 2. Inverse Perspective Projection of an Image Line.

2.3 The General Formulation of the Problem

Let us consider three linear ridges L_{01}, L_{02}, L_{03} of an object model defined in a model reference system (S_{0m}) and three image lines l_{01}, l_{02}, l_{03} defined in an image reference system (S_{0u}).

We want to find, when it is possible, the rotation R_{abc} and the translation T_{uvw} to apply to the model so that the lines l_{0i} will correspond, respectively, to the perspective projections of the lines L_{3i} which are the transformed ones of the lines $L_{0i.}$

This problem can be decomposed in two parts: at first, the search of the rotation R_{abc} and in a second step the search of the translation T_{uvw}. The rotation matrix can be written as the product of three elementary rotations around the x, y, z axis (see eq. 2) [Styl00].

The components of the direction vectors N_{3i} of line L_{3i} are given by the matrix product in equation 3 and must verify the equation 5. Thus, we have to solve the three-unknown system, which is presented in equation 4. In this way a simulation of camera's taken actual situation (i.e. *interior* and *exterior* orientation) is achieved [Str94]. The solution of this system is too complicated. To simplify it, the three linear ridges of the object model must be mutually parallel or perpendicular constructing rectangular shape images, and a new *model* and *viewer* coordinate system must be set [Dhom89].

Nowadays, commercial CAD software provides a *virtual camera* in accordance with easily established user-defined *model* and viewer coordinate systems, facilitating eq. 4 solution by the proposed system. This implies some mental processes (e.g. in Top View: object's image appears to be a horizontal line; in Right and Left View: object's image appears to be a vertical line), which supports object pose estimation and an interactive user control of the spatial orientation of the object (see next Section).

$$\left[R_{\alpha\beta\gamma}\right] = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \Rightarrow$$

$$\left[R_{\alpha\beta\gamma}\right] = \begin{bmatrix} \cos\gamma\cos\beta & \cos\gamma\sin\beta\sin\alpha - \sin\gamma\cos a & \cos\gamma\sin\beta\cos\alpha + \sin\gamma\sin\alpha \\ \sin\gamma\cos\beta & \sin\gamma\sin\beta\sin\alpha + \cos\gamma\cos a & \sin\gamma\sin\beta\cos\alpha - \cos\gamma\sin\alpha \\ -\sin\beta & \cos\beta\sin\alpha & \cos\beta\cos\alpha \end{bmatrix} \qquad \text{(eq. 2)}$$

$$\begin{bmatrix} A_{3i} \\ B_{3i} \\ C_{3i} \end{bmatrix} = \left[R_{\alpha\beta\gamma}\right] \times \begin{bmatrix} A_{0i} \\ B_{0i} \\ C_{0i} \end{bmatrix} \qquad \text{(eq. 3)}$$

$$\begin{bmatrix} A_{3i} \\ B_{3i} \\ C_{3i} \end{bmatrix} \times \begin{bmatrix} b_{0i} \\ -a_{0i} \\ d_{0i}/f \end{bmatrix} = \left[R_{\alpha\beta\gamma}\right] \times \begin{bmatrix} A_{0i} \\ B_{0i} \\ C_{0i} \end{bmatrix} \times \begin{bmatrix} b_{0i} \\ -a_{0i} \\ d_{0i}/f \end{bmatrix} = 0 \quad \left\{ \begin{matrix} i = 1,2,3 \\ \\ unknown \quad \alpha,\beta,\gamma \end{matrix} \right\} \qquad \text{(eq. 4)}$$

3. THE CAMERA-VIEW PROJECTION SYSTEM

The camera-view projection system introduced in this paper profits from CAD environment facilities like *design cubes* and *virtual cameras*, trying to match the shearing photograph model by changing the position and orientation of the virtual camera whilst the actual object 3-D model is fixed.

3.1 The Perspective Projection (Camera-View Projection)

The Perspective projection, known as well as Camera-View projection, enhancing realism of modeling and mimics the way a conventional camera works. The geometry of photography is essentially equivalent with perspective projection geometry. In camera-view projection, the place of the eye being taken by virtual camera's lens, and the plane of the image corresponding to camera's plate or film. The "difference" of perspective and camera-view projections is that for the latter the '*eye-point*' (lens) is between the object (facade) and the image plane (film), and as a consequence the image is formed upside-down. Otherwise, these two projections are geometrically similar [Vic02].

3.2 The Projection Relationships between the Camera, the taken Photography and the 3-D Object (the façade case)

Using the *central projection* logic, which is the base for the camera-view projection system, there is always just <u>one particular perspective</u> regarding the camera position, the taken photo and the spatial 3-D object. In other words, in central projections a 3-D object-image has been taken using a particular interior and exterior orientation for the camera used. In practice, thanks to projection relationship between the 3-D object and the taken photo, there are as many as <u>eight interior and exterior orientations</u> of the camera used [Pat02] that could be lead to a particular image (photo). Four of these orientations are related for transparent object-facade visualization and can be reduced. The selection of the remaining four orientations is a problem, that could be avoided if approximating values, taken empirically, are used in exterior orientation; whilst the interior orientation values are known from an existing documentation.

4. APPROXIMATE POSITIONING OF A VIRTUAL CAMERA

General graphics programming packages provide an extensive set of graphics functions that can be used in a high level programming language. Such as examples are the GL (Graphics Library) system on Silicon Graphics equipment, Bentley's MDL on MicroStation PC platform and the graphics standards GKS and PHIGS+. On these environments customised software extensions it is possible to deal with a digital virtual camera.

In the proposed system, by graphical means and programming techniques, it is possible to “position” and “aim” the *virtual camera* anywhere into the 3-D design session space. In this CAD system there is a *virtual camera* associated with each view and this camera can be turned on or off at will. Various settings associated with this camera allow a user to vary the type of lens and the camera’s exterior orientation, i.e. the position of the camera in current design session according to design cube and where the camera is aimed.

The projection used is a **camera-view projection**, and the camera position (*Xo, Yo, Zo*) and orientation (Up-vector or ω, φ, κ), for this particular view, can be found precisely without using control points.

The camera can be placed anywhere in the design session, looking in any direction. In addition, by moving the camera and its target point progressively through a model, while the rendered views are saved each time, a ”walkthrough” presentation of the 3-D object is produced. The whole operational procedure follows the next steps: First a screen-window segmentation is performed, and then the placement of both models (TARGET and CAD) is followed.

The placement of the TARGET model is done in red colouring (*red TARGET model*) according to provided image co-ordinates (pixel co-ordinates) for the four facade's vertices. For this purpose a low-cost image processing software could be used.

The placement of the CAD model is done in black colouring (*generic CAD model*) according to any available measurements for facade's edges (e.g. real ground co-ordinates in cm). The facade object must be orthogonal and planar. The design session used has m, cm and pu-positional units (pixels) as working units, with resolution 1cm=10 pu. The *virtual camera* is placed by default in front of the planar facade in a vertical distance according to eq. 5.

$$Yo = S * f = A * f / a$$

Where: *S: the scale of the photograph; f: the focal length; A, a: the distances of a facade edge in ground and photo respectively.*

The CAD-based procedure for camera position approximation, is based on adjustment methods trying to minimise the differences between the "**red TARGET model**" and the "**generic CAD model**"; whilst the virtual camera is moved into the design session. For this purpose an automatic and a manual approach used in turn [see Fig. 4, 5, 6 and 7].

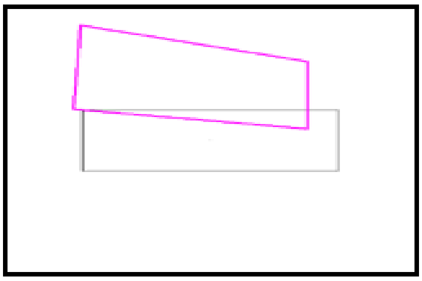

Figure 3. Red TARGET Model and Generic CAD Model in initial state before Rectification.

4.1 The Semi-Automatic Approach

This approach is empirical and it is based on attempts for approximating camera position and orientation. It is well known that the optical-mechanical rectification procedure has to be performed in three steps [Styl99]. Similarly, the empirical automatic procedure follows these steps in a semi-automatic way. Three push-buttons are available in application's GUI for these steps (see Fig.4).

At first, the *red TARGET model* is magnified, so that to coincide the generic CAD model on the two upper vertices. At second, the *generic CAD model* is rotated about the X-axis, so that both down edges are equal in length. During this step some manual operations in the generic CAD model scale are needed, to ensure that the two upper vertices coincide all the time. Finally, the *generic CAD model* is rotated about Y-axis, so that both right edges are equal in length. During this automatic procedure the current values of *Xo, Yo, Zo,* ω, φ, and κ are displayed for mental monitoring and feedback purposes.

4.2 The Manual Approach

In the manual approach push-buttons control micro-rotations for ω, φ, κ and micro-movements for *Xo, Yo,* and *Zo* according to a user-defined step. These window-buttons allow changes to be made to the six degrees of freedom for the *virtual camera* separately. During both approaches the topological differences of both models are recorded and these differences define the taken accuracy of the approximation procedure. When this accuracy is not adequate, new approximated values are calculated and used as feedback to an adjustment algorithm until a predefined accuracy is achieved. The adjustment algorithm positions in turn the virtual camera whilst the aim, that is the “red TARGET model”, is fixed. During this procedure when an attempt leads to model differences smaller than a given accuracy, the camera position and camera settings' parameters are displayed and saved for future use.

These final values actually approximate the exterior orientation parameters of the camera used to take the particular photo. The camera settings (FoV, f) used in this procedure are fixed and they could be found empirically or they are known. The **virtual camera** used in CAD environments has some advantages over **conventional cameras**. Hence, in virtual cameras: a) Everything is in focus; no matter how close to or far from the camera, b) There are no problems related with depth of fields, astigmatism, curvature of field and aberrations.

4.3 The Graphical User Interface (GUI)

A GUI dialog box, with a number of push-buttons, provides a user-friendly interface and powerful event-driven functions for facade's modeling (Fig. 4).

The dialog box event-driven controls are related to both models (red TARGET, generic CAD) and they are performed in a particular screen view; whilst the remaining views used for facade's visualization.

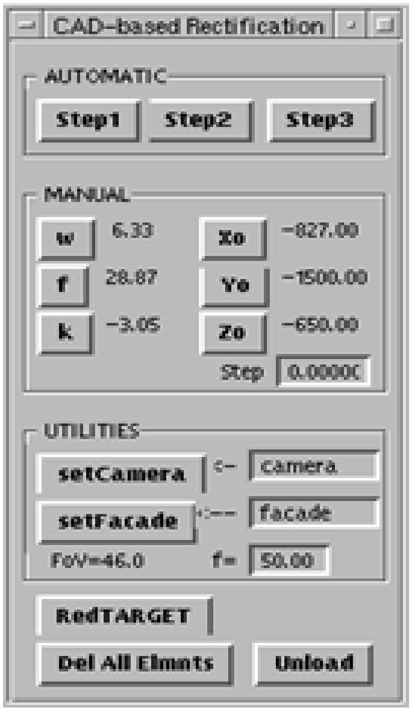

Figure 4. The Dialog Box (GUI) used to control Exterior Orientation Rectification.

This dialog box is divided in three parts: The first is devoted to the **Automatic** procedure, the second to **Manual** enhancements, and the third to some useful **Utilities** like "*Delete All Elements*", "*Unload Application*" and so on.

Interfaces are designed to carry on a continual interactive dialogue for current values concerning *virtual camera's* exterior orientation, so that the user is informed of actions in progress at each step. This is particularly important for feedback purposes in a mental procedure. Good diagnostics and error messages are designed to help determine the cause of an error. The Figure 4 demonstrates the dialog box used to control the process [Styl02].

5. POSE DETERMINATION EXPERIMENTS -LINE MATCHING

5.1 A Case Study – Practical Example

The developed CAD-based method has been verified practically on a neo-classical historical building. This building, built in 1934, is situated in Melbourne-Australia and the rectification parameters, of an available facade's photograph, are known by analytical photogrammetric enginnering methods. These parameters were used in a cross-reference test concerning an edge-distance (see Table 2).

The wide-edge of the orthogonal, planar and discrete facade was 7.5 m and the height-edge was 2.8 m. Using PhotoShop and the available photograph of the facade, the image co-ordinates of the four corners (in pixels) were found as (x/y): 92/153, 373/221, 373/348, and 88/311. The design file working units were in m, cm and positional units (pixels) with resolution: 1 m = 100 cm = 1000 pu.

Initially, the *virtual camera* was at the ideal position just over the facade, with co-ordinates: Xo = 0.0, Yo = -1400.0 mm, Zo = 0.0 and focal length: f = 50 mm. Using the *automatic* and *manual* approaches, described in Section 4, the final best fitting *virtual camera* position, for the particular photo, found as: Xo = -9.78 m, Yo = -11.36 m, Zo = -3.19 m, with the angular orientation parameters as: ω = 12.71 dgrs, φ = 40.32 dgrs, κ = -8.18 dgrs. The focal length of the *virtual camera* producing this coincidence found as f = 64.1 mm (with a Field Of Vision = 36.6 dgrs).

On Figure 5, facade's models are displayed as they are just after the first step of the automatic procedure. On Figure 6, facade’s models are displayed as they are in a middle-procedure stage, and particularly after the second step of the automatic procedure. On Figure 7, both models (red TARGET and generic CAD) are displayed in a coincide state after the proposed in this paper CAD-based rectification.

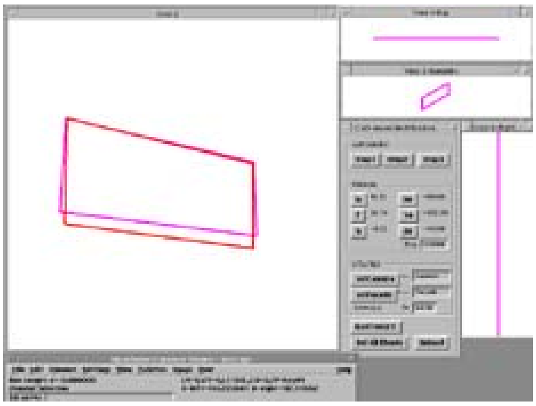

Figure 5. Red TARGET Model and Generic CAD Model after Step1, Automatic Approach (f=64.1 mm).

(ω = 14.72 degrees, φ = 34.74 degrees, κ = -8.33 degrees, Xo = -8.89 m, Yo = -12.82 m, Zo = -4.10 m)

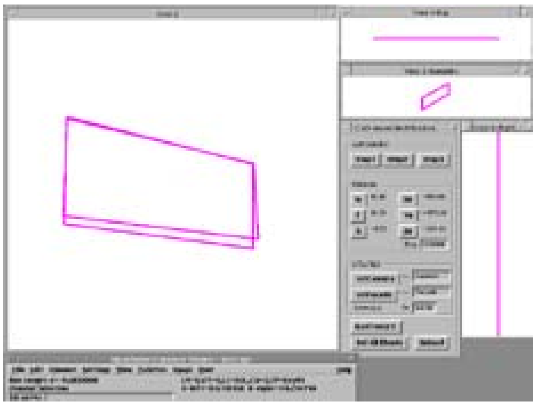

Figure 6. Red TARGET Model and Generic CAD Model after Step2, Automatic Approach (f=64.1 mm).

(ω = 14.96 degrees, φ = 38.75 degrees, κ = -8.53 degrees, Xo = -9.50 m, Yo = -11.70 m, Zo = -3.94 m)

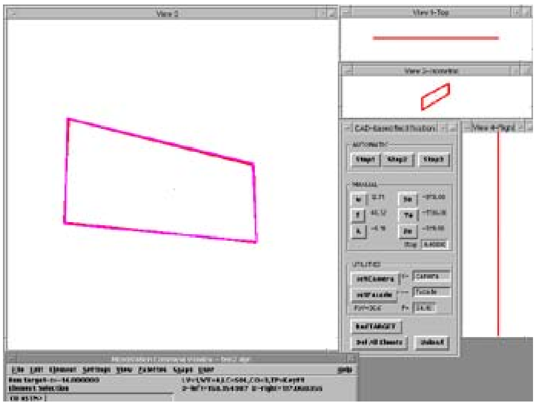

Figure 7. The Final Image after Rectification (red TARGET model = generic CAD model) (f = 64.1 mm).

(ω = 12.71 degrees, φ = 40.32 degrees, κ = -8.18 degrees, Xo = -9.78 m, Yo = -11.36 m, Zo = -3.19 m)

5.2 Accuracy Evaluation

The developed system has been verified using as test-object the facade of Section's 5.1 case study. Table 1 shows *rectification parameters* and *rotation matrix elements* produced by the proposed method.

CAD-based Rectification Method	
Rectification Parameters	**Rotation Matrix Elements**
a1= -0.185066	R11= 0.754705
a2= -0.156449	R12= 0.002038
a3= 0.013760	R13=-0.656061
b1= 0.018976	R21= 0.108531
b2= -0.035481	R22= 0.985829
b3= -0.002683	R23= 0.127912
d1= -7.005009	R31= 0.647025
d2=-12.079406	R32=-0.167739
	R33= 0.743789

Table 1. Rectification Parameters.

The accuracy of this CAD-based rectification was determined by comparing an object's edge-distance calculated using the eight rectification parameters produced by this system, with the similar edge-distance calculated by the classical analytical photogrammetric method (*collinearity method*).

Table 2 shows values of *exterior orientation parameters* and the facade's edge-distance found using both the proposed CAD-based method and the collinearity method. The Inner orientation was: c = 64.1 mm, xo = 0.0 mm, yo = 0.0 mm.

Exterior Orientation Parameters (CAD-based Method)			
Xo= -9.78 m	ω =12.71degrees		
Yo=-11.36 m	φ =40.32degrees		
Zo= -3.19 m	κ =-8.18degrees		
Lengths	**CAD-based**	**Collinearity**	**Difference**
Horizontal	801.902 cm	801.105 cm	0.797 cm
Vertical	853.298 cm	852.590 cm	0.708 cm

Table 2. Accuracy Evaluation.

The pose determination error is less than 10^{-3}, so the approach is suitable for terrestrial photogrammetric applications, mobile robot tracking, etc.

6. RULES & PRINCIPLES OF THE LINE MATCHING PROCEDURE

The development of this semi-automatic digital rectification has the following principles:

- Interactive manipulation of both facade's models (*red TARGET* and *generic CAD*) for an on-screen CAD-based rectification.
- Automatic and manual procedures for this rectification with a friendly easy-to-use GUI.
- Analytical rectification support (i.e. automatic calculation of rectification parameters).

- Stereo rectification support (i.e. performing the bundle adjustment for subsequent 3-D modeling) resulting in object pose determination.
- As-built facade's 3-D modeling using design utilities from the supporting CAD platform.
- Modeling accuracy 10^{-3} metric units, suitable for Architectural and Archaeological applications.
- Educational teaching-photogrammetry utilities, for on-screen real time rectification.
- Open-design logic for possible future extensions (e.g. regarding CAD/VR/multimedia integration).

7. CONCLUSIONS

The proposed <u>interactive</u> method for pose accurate determination from a single image, may be a useful alternative to popular single image pose algorithms. It does not require an initial pose estimation for accurate determination and it can be run faster than the <u>iterative</u> and <u>closed-form</u> solutions since it typically requires an order of magnitude fewer arithmetic operations (according to timing results).

The accuracy of the proposed method is about 10^{-3} in any metric unit. Using better imaging devices, employing sub-pixel preprocessing techniques and performing more accurate focal length calibrations, the proposed method will increase the pose determination accuracy. Also, pose accuracy can be improved if multiple rectangular shapes are manipulated (e.g. in city modeling); provided that the relative positions and orientations of each rectangular shape are known in advance. Future research may be conducted to extending the location approach to outdoor applications, to applying the proposed approach to mobile robot guidance to determing pose by using planar curves on a model object surface, and to constructing virtual reality hardware for real-time pose determination. The proposed CAD-based system provides a first step towards the integration of photogrammetric and robotics knowledge for an exterior orientation parameters' evaluation without control points.

Future enhancements will be on using interactive virtual-reality environments, that is headsets and data gloves, or even stereoscopic glasses and video monitors. In these environments, as the head-eye position changes the viewing position for the object's facade is changed as well. For conservation purposes and in a meta-documentation logic, camera positioning for particular documented photos can be approximated and then to be used for 3-D modeling in any CAD environment. This method is well operated when an existing meta-documentation for a photograph offers the interior orientation parameters and qualitatively object information, like: "*every side is parallel to the opposite one*", "*facade's edges are equal in length and perpendicular*", etc.

8. ACKNOWLEDGMENTS

This research was supported by the University Committee on Research at the ATEI of Thessaloniki, Greece, for the project: "*Digital Documentation of Monuments and Sites*", 1998-2002.

9. REFERENCES

[Abid96] Abidi, M.A., and Chandra, T.A. New Efficient and Direct Solution for Pose Estimation Using Quadrangular Targets, Algorithms and Evaluation, Vol. 18. 1992.

[Alv02] Alvarez, B.S.V., and Carvallo, P.C.P., Gattass, M., Insertion of Three – Dimensional Objects in Architectural Photos. Journal of WSCG, Vol.10, No.1., 2002.

[DeM95] DeMenthon, D.F., and Davis, L.S. Model-Based Objects Pose in 25 Lines at Code. International Journal of Computer Vision, Vol.15, pp. 123-141, 1995.

[Dhom89] Dhome, M., Richetin, M., Lapreste, J.-T., and Rives, G. Determination of the Attitude of 3-D Objects from Single Perspective View. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.11, No.12, pp.1265-1278, 1989.

[Pat02] Patias, P., Tsioukas, V., Sechidis, L., A new Digital Filter for Extracting of Conjugate Points on Epipolar Stereo Images, ISPRS, Corfu 2002.

[Smin02] Sminchisescu, C., and Telea, A., Human Pose Estimation from Silhouettes - A Consistent Approach Using Distance Level Sets. Journal of WSCG, Vol.10, No.1., 2002.

[Str94] Streilein, A. Towards Automation in Architectural Photogrammetry: CAD-Based 3D-Feature Extraction. ISPRS Journal of Photogrammetry & Remote Sensing, Spring, 1994.

[Styl99] Styliadis, A., Patias, P., and Paraschakis, I. CAD-Supported Determination of Sensor Attitude in Terrestrial Photogrammetric Applications. In: International Archives of Photogrammetry and Remote Sensing, Vienna, Austria, Vol. XXXI, Part B5, pp.451-456, 1999.

[Styl00] Styliadis, A. (ed.). Computer Graphics – Algorithms & Programming. Ziti Editions, ISBN 960-431-510-2, pp. 416, 2000.

[Styl02] Styliadis, A. (ed.). Programming the User Interface in Human-Computer Interaction – A Computer GIS Perspective. Ziti Editions, ISBN 960-431-768-7, pp. 484, 2002.

[Vic02] Vicente, M.A., Gil, P., Reinoso, O., and Torres, F. Objects Recognition by Means of Projective Invariants Considering Corner –Points. Journal of WSCG, Vol.10, No.1., 2002.