JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 4, September 2007

Evaluation of Machine Vision Algorithms for Autonomous Aerial Refueling for Unmanned Aerial Vehicles

M.L. Fravolini*
University of Perugia, 06100 Perugia, Italy

G. Campa† and M.R. Napolitano‡
West Virginia University, Morgantown, WV 26506

DOI: 10.2514/1.17269

The use of a combined Machine Vision (MV) and GPS-based approach has been recently proposed in simulation efforts as an alternative approach to 'pure GPS' for the problem of Autonomous Aerial Refueling (AAR) for Unmanned Aerial Vehicles (UAVs). While MV has appealing capabilities, a few critical issues need to be addressed for the actual implementation of MV for the AAR problem. For this purpose a simulation environment was developed featuring an interaction with a 3D Virtual Reality (VR) interface that generates an image stream of the AAR maneuver. The image flow is processed by the MV algorithm, providing, as output, a vector of the estimates of the relative tanker-UAV distance and attitude. This signal is then used by the UAV feedback control laws for 'tracking & docking' to the refueling boom. The MV algorithm specifically provides image processing for the isolation of optical markers, which are located at specific points on the tanker, extraction of the marker center of gravity, marker matching algorithm, and pose estimation algorithm for the final evaluation of the relative distance vector. Within this effort emphasis was placed on the development of an 'ad-hoc' feature matching algorithm followed by a comparative analysis of the performance of different matching algorithms. The paper presents a detailed analysis of the results from open loop and closed loop simulation of the different MV algorithms.

Nomenclature

3DW	3 dimensional window
AAR	autonomous aerial refueling
CCD	charged coupled device
CG	center of gravity
DFG	digital frame grabber
FM	feature matching
GLSDC	Gaussian least squares differential correction
j	index of a generic marker
k	sampling time

Received 20 April 2005; revision received 07 January 2007; accepted for publication 05 June 2007.

* Research Assistant Professor, Department of Electronic and Information Engineering, University of Perugia, 06100 Perugia, Italy.
† Research Assistant Professor, Department of Aerospace Engineering, West Virginia University, Morgantown, WV 26506/6106, USA
‡ Professor, Department of Aerospace Engineering, West Virginia University, Morgantown, WV 26506/6106, USA

LQR	linear quadratic regulator
m	number of mv detected markers
m_L	number of labeled markers
MV	machine vision
N	number of physical markers
PE	pose estimation
S_{ij}	matching support
STD	standard deviation
VRT	virtual reality toolbox

I. Introduction

ONE of the biggest current limitations of UAVs is their lack of autonomous aerial refueling (AAR) capabilities. To achieve AAR capabilities specific technical challenges need to be overcome. For AAR purposes, a key issue is represented by the need of a high accuracy measurement of the relative 'Tanker-UAV' position and orientation in the final phase of docking and during the refueling. Although sensors based on laser, infrared radar, and GPS technologies are suitable for autonomous docking,[1] the use of MV technology has been recently proposed in addition or as an alternative to these technologies.[2,3,4] Furthermore, a MV-based system has been proposed for close proximity operations of aerospace vehicles[5] and for the navigation of UAVs.[6]

The estimation of the 3D orientation and position of an object from its images (*pose estimation*) is a well-known topic in Computer Vision research. Within this effort, the 'Pose Estimation Problem' has been considered for the purpose of evaluating the relative 'Tanker-UAV' position and orientation. The control objective is to guide the UAV within a defined 3D Window (3DW) below the tanker where the boom operator can then manually proceed to the docking of the refueling boom with the UAV fuel receptacle followed by the refueling phase. A digital camera installed on the UAV generates a stream of images of the tanker. These images are then processed by a set of MV algorithms providing in real time an estimate of the tanker-UAV relative position and orientation. This problem is well known in the technical literature as the 'Pose Estimation' problem. The set-up of the pose estimation is substantially improved if a number of features can be detected from the digital images of the tanker.

For the purpose of this study these features are selected to be optical markers, which can be considered to be active sources of lights. It should be emphasized that this assumption does not subtract from the generality of the approach. In fact, different features could be selected—such as passive source of lights or even tanker geometric corners—without any loss of generality.

The image processing algorithms have to be computationally simple, reliable and suitable for real-time implementation. Within the pose estimation algorithm it is assumed that between sequential images a set of corresponding points is always found and available. In other words, it is assumed that the position of the detected markers in the object reference frame is known.

It is clear that severe problems may arise when this assumption is violated, leading to potentially biased estimation of the relative position parameters.[7] This problem is typically present when the number of corresponding points (that is, the number of detected markers) is small and time varying due to possible physical occlusions and/or loss of visibility during the approach and docking phases. For example, as the UAV approaches the tanker, the presence of the boom itself could lead to a loss of visibility of one or multiple markers. Therefore, within this effort emphasis has been placed in studying the feature correspondence problem interfaced to the pose estimation algorithm within the AAR problem.

A detailed AAR simulation environment has been developed by the authors in[8] with the UAV modeled with the parameters of the ICE-101 aircraft[9] and the tanker modeled with the parameter of a KC135 aircraft.[10] This simulation provides detailed modeling for the elastic flexibility of the boom, the wake effects from the tanker on the UAV, and the atmospheric turbulence. The 'tracking & docking' control laws have been designed using a conventional LQR-based approach.

With respect to the previous effort by the authors, the simulation environment has been augmented with a Virtual Reality interface providing a stream of images. These images are then used by a MV algorithm for the final derivation of the 'tanker-UAV' relative distance vector. The reliability of the pose estimation provided by the MV algorithm,

based on different feature matching techniques has been analyzed. Extensive open loop and closed loop simulation results of the various MV algorithms have also been analyzed and discussed.

II. The MV-based AAR Problem

A block diagram of the MV-based AAR problem is shown in Fig. 1. The relevant geometric distances and associated reference frames are also shown in the figure.

A. Reference Frames

The study of the AAR problem requires the definition of the following Reference Frames (RFs):

- E_{RF} is the earth-fixed reference frame
- T_{RF} is the body-fixed tanker reference frame located at the aircraft center of gravity
- U_{RF} is the body-fixed UAV reference frame located at the aircraft center of gravity
- C_{RF} is the body-fixed UAV camera reference frame.

The following notations will be used in this paper. AB^e represents the vector from point A to point B expressed within the E_{RF}; the matrix R_t^e represents the rotation of the T_{RF} with respect to the E_{RF}. To make the docking problem invariant with respect to the nominal heading of the aircraft, an additional fixed frame M_{RF} is defined. This frame is rotated of the nominal heading angle Ψ_0 with respect to the E_{RF}. The transformed vectors in the M_{RF} are evaluated using the relationships: $RB^m = R_e^m \cdot RB^e$ where R_e^m is a fixed rotation matrix from E_{RF} to M_{RF}.

B. Problem Formulation

The objective is to guide the UAV such that its fuel receptacle (point R in Fig. 1) tracks the center of a 3-dimensional window (3DW) under the tanker (point B). Once the UAV fuel receptacle reaches and remains within this 3DW, the boom operator is assumed to take control of the refueling operations. It should be underlined that point B is fixed in the T_{RF} with the dimensions of the 3DW $(\delta x, \delta y, \delta z)$ being an important design parameter. It is assumed that the tanker and the UAV share a data communication link. The UAV is equipped with a digital camera along with MV algorithms acquiring the images of the tanker and extracting the point-wise features (points P_j) from the tanker image. For simplicity, the 2-D image plane of the MV is assumed to be coincident with the 'x–z' plane of the C_{RF}.

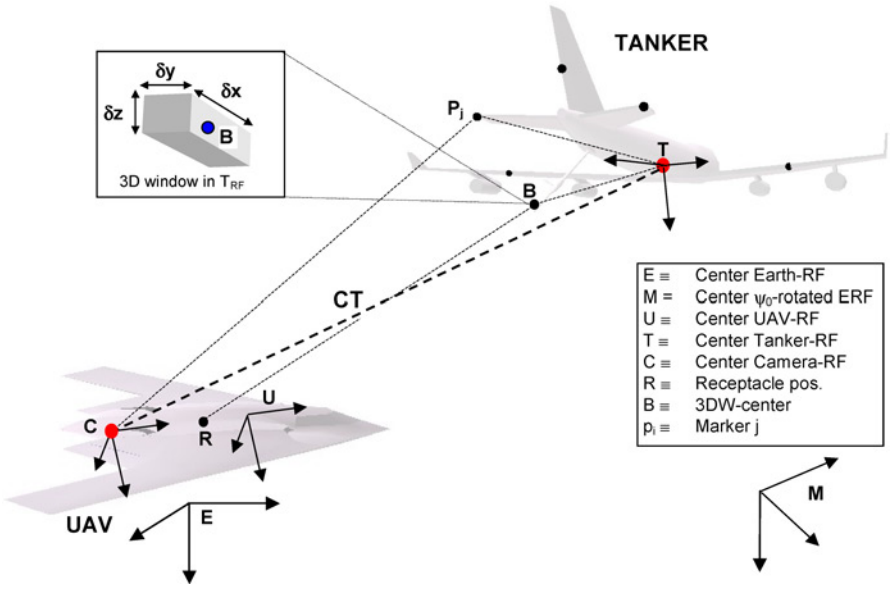

Fig. 1 Block diagram of the AAR problem with relevant reference frames.

C. Receptacle-3DW-Center Vector

As described above the reliability of the AAR is based on the accuracy in the measurement of the distance RB^e. This distance is derived from the available measurements of distance and attitude coming from the UAV and the tanker. For this purpose, the following geometric transformation is introduced:

$$RB^e = R_u^e \cdot \left(UC^u - UR^u + R_c^u \cdot CT^c\right) + R_t^e \cdot TB^t \tag{1}$$

Since the fuel receptacle and the 3DW-center are located at fixed and known positions with respect to center of gravity of the aircraft, in the above equation the vectors UR^u and TB^t are exactly known, while the values of rotation matrix R_t^e and R_u^e are derived by the attitude sensors of the tanker and the UAV respectively. The camera-tanker distance (CT^c vector) expressed along the C_{RF} is provided by the MV algorithms. The accuracy of the estimation of the CT^c vector will be extensively discussed in this paper.

III. The Virtual Environment for AAR Simulation

A. Tanker and UAV Dynamics

The aircraft models used for AAR simulation purposes have been developed using the conventional modeling approach outlined in.[11] The resulting model is described by a (12 states) state space model. First order dynamic models have been used for the modeling of the actuators using typical values for aircraft of similar size and/or weight. The B747-tanker and UAV features a typical set of autopilot systems designed using aconventional LQR approach. Details on the design of the tracking and docking control scheme are provided in.[8] The boom system has been modelled using a 3-D finite elements model.[12] The atmospheric turbulence acting on the refueling boom and on both aircraft has been modeled using the Dryden wind turbulence model.[13] A 'light' turbulence was selected since aerial refueling is typically performed at high altitudes in calm air. The wake effects of the tanker on the UAV are more significant than the atmospheric turbulence and have been modeled through the interpolation from a large amount of experimental data[14,15] as perturbations to the aerodynamic coefficients $c_D, c_L, c_m, c_Y, c_l, c_n$ for the UAV aerodynamic forces and moments.

B. The 3D Visual Interface

The modeling described above was linked to a Virtual Reality Toolbox® (VRT) interface to provide the 3D graphics associated with the AAR maneuvers. Specifically, the interface allows the positions of the simulated objects, to drive the position and orientation of the corresponding objects in a Virtual World. Within this Virtual World, optical markers, were located on the B747 tanker model at specific locations, as discussed in.[8] The markers were modeled as red spheres with a radius of 10 cm each. The images of the tanker were then acquired and processed by a 'virtual' set of MV algorithms, which in real-life are to be hosted in the UAV flight computer.

Figure 2a shows a typical VRT image as seen by the UAV camera, while Fig. 2b shows the selected positions for the markers.

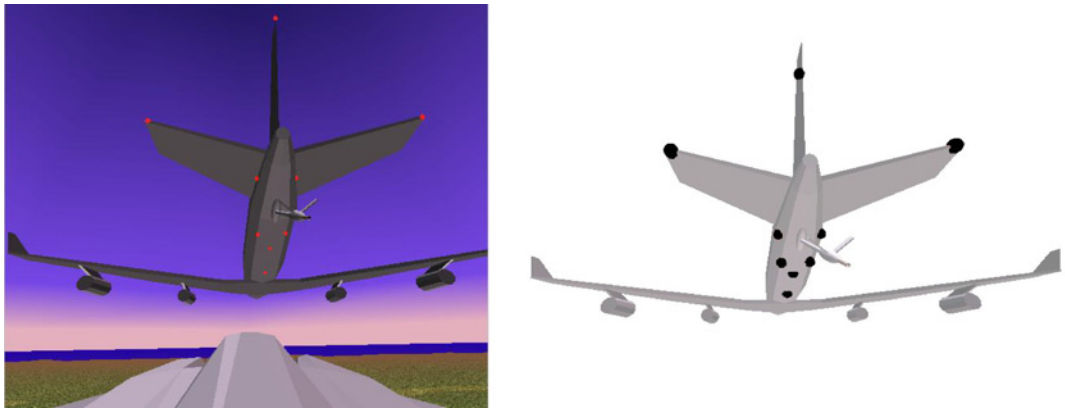

Fig. 2 a) VRT image, b) position of the optical markers.

IV. The MV System

The UAV-tanker distance vector CT^c is estimated through the MV-based algorithms in the block diagram shown in Fig. 3. This scheme, featuring the components described below, will be referred as the "realistic MV model":

- VRT interface: This block provides the continuous image stream associated with the (tanker + UAV) system dynamics generated from the simulation.
- Digital Frame Grabber (DFG): This block represents the interface between the simulated world and the MV system. Particularly, it emulates the process of the acquisition of the digital image. The DFG is characterized by the pixel resolution of the camera. In this effort a matrix of 1280×1280 pixels has been assumed with the image rate set at 20 frames/sec. The output stream produces images in bitmap format.
- Image Processing: This block performs several important functions. Digital image processing is applied to extract the 'point features' from the stream of the images. First, the image is filtered through the application of a feature detection algorithm. Since red markers have been considered in this study, the detection was performed using a simple 'red color enhancing filter' with the goal of isolating the "spots" relative to the red markers. Next, additional image manipulation tools are applied, leading, as final result, to a binary black and white image. Finally, after the image of the markers has been isolated from the background, an 'ad hoc' procedure, consisting of morphological closures and openings, is applied to isolate and label each single connected spot. A spot is considered significant only if it contains a number of pixels larger than a pre-defined threshold. All the image manipulations previously described were performed using functions of the Matlab Image Toolbox.
- CG extraction: This block computes the center of gravity (CG) of each marker projection, so that each marker projection can be represented by a single couple of u and v coordinates.
- Feature matching algorithm: The previous 'CG Extraction' block generates, for each frame, the 2D coordinates $[u_j, v_j]$ of the CG for each of the markers, which are detected in that particular frame. At this time, these points are not conceptually associated to any specific marker of the tanker. In fact, in the general case of N detected CGs for N markers, there are $N!$ possible associations between CG points and detected markers to be explored. The above problem is a classical '*Feature Matching*' (FM) problem. It is clear that this problem has major implications on the closed-loop performance of the docking control laws. The FM algorithms will be described with details in section 5.
- Pose estimation algorithm: Finally, after the set of 'detected markers' has been matched to the full set or to a subset, in case of loss of visibility, of the physical markers of the tanker, a '*Pose Estimation*' (PE) algorithm

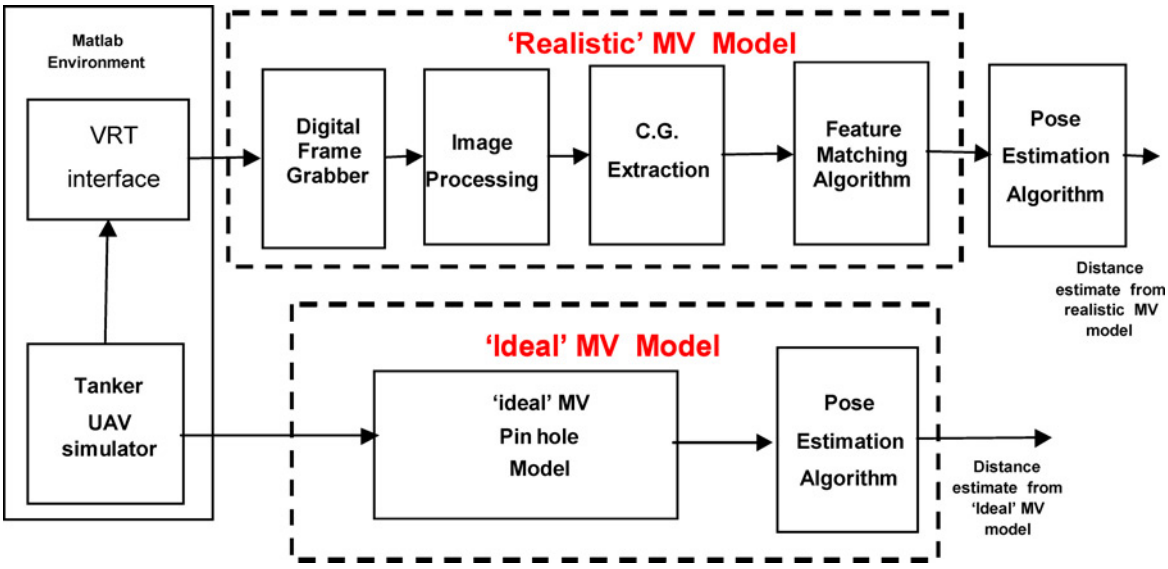

Fig. 3 Components of the 'realistic' and 'ideal' MV models.

is used to provide estimates of the UAV-tanker relative distance and pose. The PE algorithm will be described with details in section 6.

- MV 'pin hole' camera: An 'ideal' MV model, as opposed to the 'realistic' MV model with the components described above, has been also implemented using a geometric projection model ('pin hole model') for performance comparison purposes. In this model the coordinates of the markers in $T_{RF}$are projected without any error on the MV plane. In this "ideal" MV a nominal (error free) matching has also been assumed for all the detected markers. The PE algorithm is the same as for the 'realistic' MV model. The 'ideal' MV model is shown in Fig. 3.

V. The 'Image Matching' Problem

Following the extraction of the 2D coordinates $[u_j v_j]$ of the GCs of the detected markers from the image, the next task is to assign to each of them a particular optical marker P_j on the tanker, as shown in Fig. 4. The general approach is to identify a set of detected markers $[u_j v_j]$ to be matched to a subset of estimated markers positions $[\hat{u}_j\ \hat{v}_j]$ through a projection model ("pin-hole" model[21,16]) that estimates the projection in the camera plane of the nominal position of the markers P_j.

According to this model, a marker 'j' with coordinates $CP^c_{(j)} = [x^c_{p,j}, y^c_{p,j}, z^c_{p,j}]^T$ in the C_{RF} frame is projected into the image plane whose coordinates $[\hat{u}_j, \hat{v}_j]$ can be computed using the projection equation:

$$\begin{bmatrix} \hat{u}_j \\ \hat{v}_j \end{bmatrix} = g\left(f, CP^c_{(j)}\right) = \frac{f}{x^c_{p,j}} \begin{bmatrix} y^c_{p,j} \\ z^c_{p,j} \end{bmatrix} \tag{2}$$

where f is the camera focal length. The components of the marker P_j expressed in the T_{RF} are fixed and known 'a priori' with coordinates defined by the vector $TP^t_{(j)}$. The following geometric equation relates the vector $CP^c_{(j)}$ to the vector $TP^t_{(j)}$:

$$CP^c_{(j)} = CT^c + R^c_t \cdot TP^t_{(j)} \tag{3}$$

where the vector $CT^c = [x^c_t, y_t{}^c, z^c_t]^T$ and the matrix R^c_trepresents the origin and the orientation of the T_{RF} with respect to the C_{RF} respectively. Matrix R^c_t is specified by the relative yaw, pitch, and roll angles $\Phi_c = [\psi^c_t, \theta^c_t, \varphi^c_t]^{\mathrm{T}}$. The vector CT^c and the matrix R^c_t are not directly available but are derived from available measurements. Particularly,

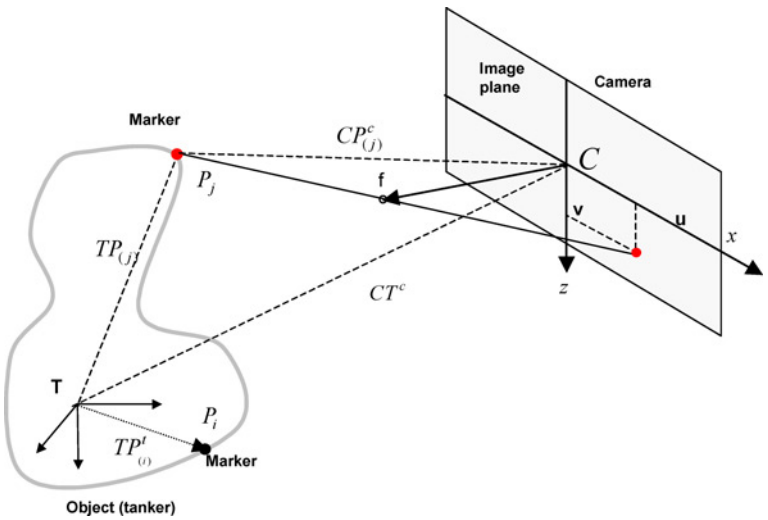

Fig. 4 The MV "pin-hole" model.

Table 1 Simulation parameters.

Heading angle: $\Psi_0 = 0°$	MV sampling rate: 20 Hz	Dryden Model
Known distance vectors:	MV pixel size: 10 μm	at light conditions
$UC^u = [-3, 0, -0.6]$ m	GPS noise-std: 1.5 m	
$UR^u = [-1, 2.5, 0]$ m	CCD size: 1.28 cm	Wake effects
$TB^t = [-22, 0, 8]$ m	GPS sampling rate: 10 Hz	(Bihrle experimental
$\mathrm{CT}^C(0) = [125\,0\,30]$ m	V noise-std: 1.5 m/s	data)
	α, β noise-std:0.0045 rad	
	p, q, r noise-std: 0.015 rad/s	
	ψ, θ, φ noise-std: 0.007 rad	
	Transmission delay: 0.05 s	

the following relationships are used:

$$CT^c = R_u^c \cdot R_e^u \left(ET^e - EU^e\right) - R_u^c \cdot UC^u \tag{4}$$

$$R_t^c = R_u^c \cdot R_e^u \cdot R_t^e \tag{5}$$

In Eq. (4), vectors ET^e and EU^e are measured through GPS sensors installed on the tanker and UAV respectively, the attitude matrices R_e^u and R_t^e are derived using yaw, pitch, and roll angle measurements of the UAV and tanker respectively, while R_c^u is constant and known. Therefore, the accuracy of the estimates for $[\hat{u}_j, \hat{v}_j]$ is strictly related to the accuracy of the measurements involved in Eqs. (4) and (5). For the purpose of achieving realistic simulation results, a nominal level of noise has been 'injected' in all the measurements involved in Eqs. (4) and (5) (see Table 1).

A. Formalization of the 'Points Matching' Problem

Initially, the set of points $p_j \equiv [u_j, v_j]$ from camera measurements are not related to the actual markers on the tanker. The problem can be formalized in terms of matching the set of points $\bar{P} = (p_1, p_2, \ldots, p_m)$ to the set of points $\hat{\bar{P}} = (\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n)$ where $\hat{p}_j \equiv [\hat{u}_j, \hat{v}_j]$. In this effort the set $\bar{P}$ represents the set of the m'to be matched' detected markers extracted by the camera measurements, while the set $\hat{\bar{P}}$ represents the set of the $n (n = 9)$ "nominal" markers estimated through eq. (2). Since the data sets $\bar{P}$ and $\hat{\bar{P}}$ represents the 2D projection of the markers at the same time instant on the same plane, an high degree of correlation would be expected. In the ideal case corresponding points would be exactly superimposed and the matching process would be trivial; however, in the presence of different sources of system and measurement noise, a matching problem has to be formalized and solved. In general, a matching problem can be formalized as an optimal pairing of points in the set $\bar{P}$ to those in the set $\hat{\bar{P}}$ such that the sum of the distances between the paired points is minimized. Fig. 5 shows a correct matching between the two data sets in a typical image from the AAR simulator.

B. Approaches to the 'Points Matching' Problem

The mapping between feature points is a key issue in several computer vision problems, such as detecting moving objectives in a scene or calculating motion parameters of the camera with respect to a moving object. A detailed technical literature describes a number of robust matching techniques for point sets.[17,18] Usually, the degree of similarity between 2 data sets is defined in terms of a cost function or a distance function derived on general principles as geometric proximity, rigidity, and exclusion.[19] These constraints can be divided in local constraints, which can be evaluated independently for each possible matching, and global constraints, which need the joint analysis of a set o potential matching.[20] The best matching is then evaluated as the result of on optimization process exploring the space of the potential solutions. As later described, a graph can also be derived such that the nodes represent feature points and links represents the distance between features; therefore, the problem can be cast as a classical assignment problem. The assignment problem is a classic example of a network flow problem on a graph. Network flow problems are Linear Programming (LP) problems for which computationally efficient algorithms (that exploit the particular structure of those problems) are routinely used.

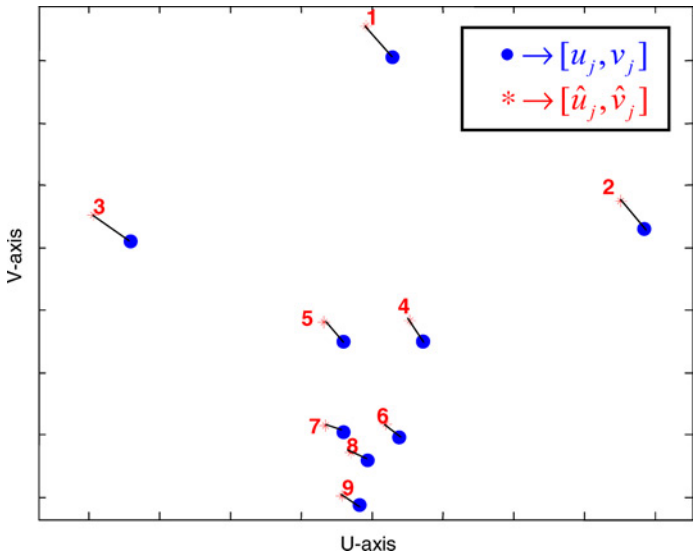

Fig. 5 Matching between the labeled set of points $\hat{\bar{P}}$ and the unlabeled set $\bar{P}$.

In[21] a comparison of different methods is shown. Interesting relaxation techniques are shown in.[22–23] Other approaches based on 'registration' and 'tracking' are shown in.[24] These approaches are often augmented by a feedback from the calculated motion parameters using matching evaluated in the previous frames.

In the AAR problem it was observed that the discrepancy between the two matching sets can be reasonably modeled by assuming a simple rigid translational model. Under this assumption the measure of the matching strength between point p_i and $\hat{p}_j$ has been defined in terms of their Euclidean distances. The distance Matrix D is defined as the matrix whose entries $D(i, j)$ are the distances d_{ij} between each couple of points 'i' and 'j' of the two sets.

The performance of a few well-known 'Point Matching' algorithms have been evaluated and compared in the context of MV-based AAR. Furthermore, a new 'ad-hoc' matching method has been here formulated, where the strength of a candidate match is measured not just in terms of the relative distance between two candidate matching points p_i and $\hat{p}_j$ but also in terms of a global matching error between all the matching points that would have been originated by applying the rigid translation associated with the two matching points p_i and $\hat{p}_j$.

C. The Bipartite Graph

A weighted bipartite graph—shown in Fig. 6—is a weighted graph whose vertex set can be divided into two disjoint sets. In this study the two sets are represented by $\bar{P}$ and $\hat{\bar{P}}$. In the bipartite graph each vertex in $\bar{P}$ is connected to all the vertex in $\hat{\bar{P}}$ trough edges; the weight d_{ij} associated to the edge $(p_i, \hat{p}_j)$ is the Euclidean distance between point p_i and point $\hat{p}_j$. The matrix $D(i, j)$ therefore contains information about all the edges in the graph. A valid matching in this graph is defined as a subset M of η edges ($\eta = \min(n, m)$) such that each vertex of the graph is incident upon at most one edge of M. Given a subset M, its weight is defined as the sum of all the η weights of M. If an edge $(p_i, \hat{p}_j)$ belongs to M, then vertex p_i is said to be matched to vertex $\hat{p}_j$ and vice versa. A matching with a minimum weight is said to be a minimum weight matching or simply a minimum matching.

A specific algorithm, called the Hungarian method, has been introduced for finding a minimum weight matching in a weighted bipartite graph. This algorithm has a complexity of $O(n^3)$. A detailed description of this algorithm can be found in.[25] The important aspect of this method is that this algorithm is always guaranteed to find the minimum weight matching. It should be emphasized that, although the Hungarian method always provides a minimum weight matching, it is not guaranteed that the provided matching is indeed correct.

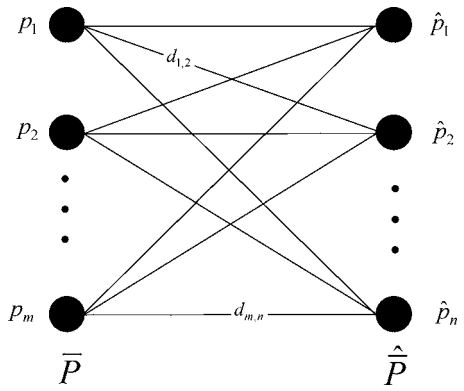

Fig. 6 Bipartite graph associated to sets $\hat{\bar{P}}$ and $\bar{P}$.

D. The Greedy Method

Within this method a greedy algorithm, outlined in[25], is used for solving the points correspondence problem. Given the graph distance matrix $D(i, j)$, the set of edges is explored to find the edge $(p_i, \hat{p}_j)$ with the smallest weight (distance d_{ij}). The point p_i is matched to the point $\hat{p}_j$; next, all the edges associated with p_i and $\hat{p}_j$ are removed from further consideration.

The above step is repeated until every point in $\bar{P}$ is matched to some point in $\hat{\bar{P}}$. Although very simple and efficient, this algorithm does not always produce a matching of minimum weight; furthermore, it is possible for the algorithm to get stuck in a local minimum.[26]

To improve the robustness of this method and to facilitate the matching, a "center of gravity correction" is preventively applied to the points of set $\hat{\bar{P}}$, to have the CG of the "translated" $\hat{\bar{P}}$ set coincident with the CG of the data set $\bar{P}$. To avoid inconsistent translations, this operation is only applied to the frames where the number of points in $\bar{P}$ and $\hat{\bar{P}}$ coincide.

E. Mixed Hungarian and Greedy Method

This method is essentially the fusion of the two previous methods. Therefore, a point matching $(p_i, \hat{p}_j)$ is considered valid if and only if the same matching is confirmed by both methods. The points p_i for which the two methods do not provide consistent results are considered unreliable and, therefore, discarded by the algorithm.

F. The Proposed Robust Matching Method

A critical issue is the robustness of the matching algorithm in presence of measurement errors affecting the estimation of the position of markers set $\hat{\bar{P}}$. Therefore, some specific features were introduced with the objective of increasing the robustness of the matching algorithm. Before describing the resulting algorithm, the following definitions are introduced:

- *Matching Support*: It is assumed that the main effect associated with the presence of the (measurement and system) noise is a rigid translation between the two matching sets. Under this assumption, for each candidate match $(p_i, \hat{p}_j)$, the vector $t_{ij} = (\hat{p}_j - p_i)$ is assumed as a possible rigid translation vector, due to the presence of the noise, from points in the set $\bar{P}$ to points in the set $\hat{\bar{P}}$. After the application of the rigid translation t_{ij} to all the points in $\bar{P}$, a generic translated point $p_h + t_{ij}$ is directly matched to the nearest point $\hat{p}_{\bar{k}_h}$ in $\hat{\bar{P}}$ according to the following "*winner takes all*" method:

$$\bar{k}_h = \arg\min_k \left(\left\| \hat{p}_k - (p_h + t_{ij}) \right\| \right) \quad k = 1, \ldots, n \tag{6}$$

The corresponding matching error is then given by:

$$\bar{e}_h = \left\| \hat{p}_{\bar{k}_h} - (p_h + t_{ij}) \right\| \quad k = 1, \ldots, n \tag{7}$$

Repeating this procedure for all the m points in $\bar{P}$, it is then possible to define a vector of possible matching $\bar{K}_{ij} = [\bar{k}_1, \ldots \bar{k}_m]$ associated to the translation t_{ij}. In the event case that two or more elements in $\bar{K}_{ij}$ receive the same index, only the association with the smallest matching error is considered valid while the remaining are deleted because their matching is considered unreliable. Therefore, the number of detected and labeled m_L markers is in general less or equal to m.

The matching support for the association set $\bar{K}_{ij}$ is then defined as:

$$S_{ij} = \frac{1}{m_L} \sum_{h=1}^{m_L} \bar{e}_h \tag{8}$$

The cost S_{ij} can then be considered a measurement of the robustness for the candidate match$\bar{K}_{ij}$, since it takes into account the overall mean matching error between the two data sets. The best matching $\bar{K}^*_{ij}$ is then selected as the one providing the minimum value of the matching support (S^*_{ij}) evaluated among all the possible translations t_{ij}:

$$S^*_{ij} = \min_{i,j} \left(S_{ij}\right) \quad i = 1, \ldots, n \quad j = 1, \ldots, m \tag{9}$$

Additional features could be introduced into the matching algorithm. For example robustness could be increased by estimating marker motion parameters in successive frames and using this information to predict the future position of a labeled marker.

VI. The Real-Time Pose Estimation Problem

After the 'Point Matching' problem has been solved, the information in the set of points $\bar{P}$ are then used to derive the rigid transformation that relates the C_{RF} to T_{RF} . This topic is well-known in the technical literature as the 'Pose Estimation' (PE) problem.[27,28] Particularly, the objective is to evaluate the mapping correspondences between the 3-D physical points (markers) expressed in the object coordinates (T_{RF}) and their 2-D projections expressed in image 'u-v' coordinates (C_{RF}). For an arbitrary number of points, the most extensively used approach for the PE is based on the application of a non-linear least-squares algorithms reducing itself to the minimization of a non-linear cost function, typically solved iteratively using some variations of the Gauss-Newton method.[27,29,30,31] In particular, the Gaussian Least Squares Differential Correction (GLSDC) algorithm has been implemented in this study.[2] This algorithm has shown to provide desirable robustness, convergence, and accuracy even in presence of quantization noise produced by the CCD matrix. The non-linear 3-D to 2-D correspondence is described in terms of the projection equation already introduced in Eq. (2). The unknowns of the problems are the components of the vectors CT^c and Φ_c in Eq. (3). For simplicity purposes these unknown have been grouped in a vector $X = [x^c_t, y^c_t, z^c_t, \psi^c_t, \theta^c_t, \varphi^c_t]^T$; this vector defines uniquely the relative distance and attitude of the UAV camera fixed C_{RF} respect to the T_{RF}.

A. The Pose Estimation Algorithm

Prior to describing the algorithm the 3-D to 2-D perspective projection equations relating the unknown vector X to the known vector $TP^t_{(j)}$ are recalled below:

$$\begin{bmatrix} u_j \\ v_j \end{bmatrix} = \begin{bmatrix} g_u(f, X, TP^t_{(j)}) \\ g_v(f, X, TP^t_{(j)}) \end{bmatrix} \quad j = 1, \ldots, m_L \tag{10}$$

By grouping the equations (10) for all the 'm_L' labeled markers, the following $1 \times 2m_L$ vector of non-linear relationships is then generated:

$$G = \left[g_{u1}, g_{v1}, \ldots, g_{um_L}, g_{vm_L}\right]^T \tag{11}$$

Next, the MV estimation error at sampling time k is calculated as:

$$\Delta G(k) = G_{meas}(k) - G\left(f, \hat{X}(k), TP^t_{(j)}(k)\right) \tag{12}$$

where $G_{meas}(k)$ is the vector of the measured coordinates of the markers on the image plane and $\hat{X}(k)$ is the current estimation of vector $X(k)$.

The updated estimation $\hat{X}(k+1)$ is provided through the application of the GLSDC algorithm:

$$P(k) = A^T(k) \cdot W \cdot A(k) \tag{13}$$

$$\Delta\hat{X}(k) = P^{-1}(k) \cdot A^T(k) \cdot W \cdot \Delta G(k) \tag{14}$$

$$\hat{X}(k+1) = \hat{X}(k) + \Delta\hat{X}(k) \tag{15}$$

where A(k) is the $2m_L \times 6$ Jacobian matrix:

$$A(k) = \left.\frac{dG(k)}{dX(k)}\right|_{X=\hat{X}(k)} \tag{16}$$

nd W is the $2m_L \times 2m_L$ covariance matrix $W = \text{diag}(1/\sigma_{u1}, 1/\sigma_{v1}, \ldots 1/\sigma_{vm_L}, 1/\sigma_{vm_L})$ of the estimation error. The initial guess $\hat{X}(k=0)$ is computed using the information provided by GPSs and aircraft sensors.

B. Modification of the GLSDC

The basic algorithm (13–16) is designed to work with a fixed number of 'm_L' markers. The following set of modifications has been introduced for handling a time varying number of markers. At the beginning of each time step the number of the labeled markers is evaluated; their number is, in general, time varying because of temporary occlusions (with the markers remaining undetected) or simply because the markers do not receive a valid label from the matching algorithm. This implies that (11) has to be modified with the appropriate number of rows; next, the dimensions and the values of the matrices A and W in (13–16) also need to be revised accordingly. It should be emphasized that a minimum of 3 markers are required by the algorithm to generate a compatible solution. If more than 3 markers are available the algorithm provides a 'Least Square' solution. The improvements associated with the described modification of the GLSDC algorithm will be evident in the simulations described below.

VII. Analysis and Results

The goal of the study was to analyze the overall performance provided by the different feature labeling schemes in conjunction with the DGLS pose estimation algorithm applied to the AAR problem. A detailed set of simulation studies was performed. During the final docking phase (from the pre-contact to contract position) the UAV and the Tanker are aligned along the same heading angle. It is assumed that the UAV fuel receptacle and the on-board camera are both perfectly aligned with the UAV longitudinal axis. A simple LQR-based approach has been used for the design of the docking control laws.[8]

The time histories of the components of the CT^c vector during the AAR maneuver are shown in Fig. 7.

In the first set of simulations the purpose was to evaluate the accuracy of the estimates of the MV-based sensors during the docking; therefore, in this case the measurements provided by the MV-based sensor are not used by the control laws, to which 'nominal' noise-free measurements are provided. In the second set of simulations the MV-based measurements of the CT vector were provided directly to the control laws. For simulation purposes, 'nominal' levels of uniformly distributed white noise were artificially added to all the measurements involved in Eqs. (4) and (5). It should be emphasized that the presence of measurement errors is the main reason for the discrepancy in the point matching which has been observed in Fig. 5. Different simulation parameters along with statistical information on the noise acting on sensors, GPS and MV systems are reported in Table 1.

A. Performance of the MV System

Specific criteria for the selection of the 'best' location for the markers were not available in the technical literature. Therefore, an empirical selection of the number of markers and their location was introduced. The main criterion was to maximize their visibility from the location of the camera on the UAV. Another criteria was to avoid the installation of the markers at specific locations on the aircraft subjected to structural vibrations (for example, wing tips). An additional criterion was to avoid the installation of the markers close to the wake of the jet engines. Another criterion was to avoid the installation of the markers in any location where their accessibility could be problematic.

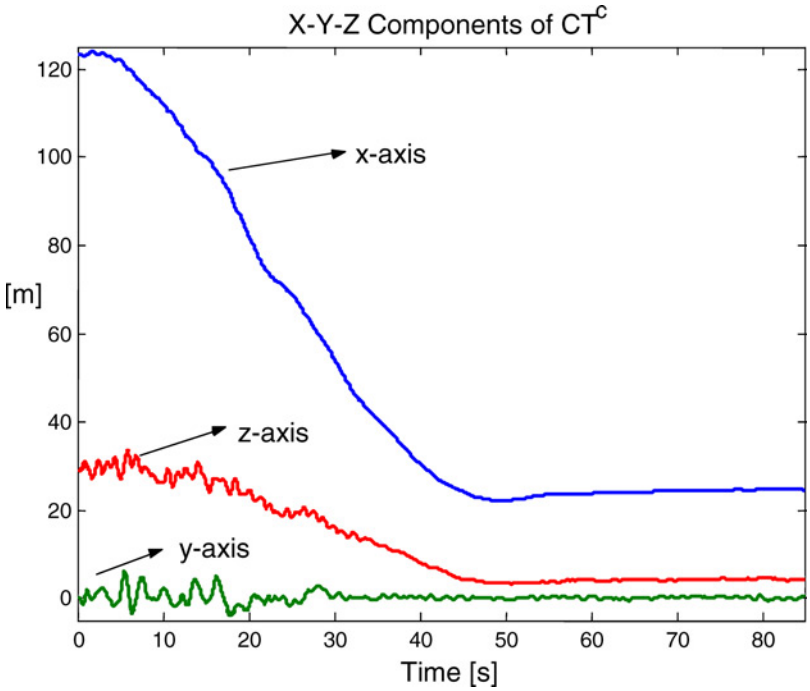

Fig. 7 Trajectory of the CT° components during the docking maneuver (CT^c is the distance between the UAV camera and the tanker's center of gravity).

A final criterion was to select locations identifying multiple planes along the x-axis of the tanker allowing, therefore, adequate levels of resolution for the acquisition of an accurate UAV-Tanker longitudinal distance.

At the end of this analysis, a set of 9 markers was selected; their location is shown in Fig. 2. This set provided sufficient redundancy and exhibited desirable robustness in the event of temporary/permanent loss of visibility by one or more markers, as described below. The effects of the camera digital resolution on the estimation error and the effects of uncertainties of the markers positions on the tanker have been described in.[32]

B. Performance of the Estimation System

As stated above, in this first study the MV-based estimates are not provided to the docking control laws. The norm of the measurement error $e_{MV} = CT^c - CT^c_{(mv)}$ was evaluated to quantify the error:

$$|e_{MV}| = \sqrt{\left(CT^c_x - CT^c_{(mv,x)}\right)^2 + \left(CT^c_y - CT^c_{(mv,y)}\right)^2 + \left(CT^c_z - CT^c_{(mv,z)}\right)^2} \tag{17}$$

where CT^c is the ideal signal and $CT^c_{(mv)}$ is the corresponding provided by the MV. The following 5 estimation algorithms for the evaluation the tanker/UAV relative distance vector have been compared:

A). "Ideal MV" model, implying a 'perfect labeling' algorithm with an 'ideal MV' featuring ideal pointwise markers without physical occlusions. The DGLS algorithm is then used for the solution of the 'pose estimation' problem. The purpose of this scenario is to provide a benchmark.

B). "Realistic MV model B", including 'digital frame grabber' and CG extraction procedures. This scheme features the 'robust labelling method', described above, and the DGLS algorithm for the pose estimation' problem.

C). "Realistic MV model C": as in case B) with the '*Hungarian'* labelling method in lieu of the 'robust labelling method'.

D). "Realistic MV model D": as in case B) with the '*Mixed'* labelling method in lieu of the 'robust labelling method'.

E). "Realistic MV model E": as in case B) with the '*Greedy'* labelling method in lieu of the 'robust labelling method'.

C. Analysis of the $|e_{MV}|$ Signal

Different aspects have been evaluated and compared. Figure 8 shows the norm of the measurement error $|e_{MV}|$ vs. the norm of the distance RB^m for the 5 scenarios outlined above. The data are relative to a complete docking maneuver lasting 300 s. As described above, scenario A) provides the benchmark performance. For the more realistic scenarios *B, C, D*, and *E*, the value of $|e_{MV}|$ is the sum of the 3 specific contributions.

The first source of error is due to the quantization error associated with the digital camera. This error is more relevant at large distances and it is responsible of a biasing term (function of the distance) in the estimation. This effect becomes negligible at short distances where the estimation is nearly unbiased. The second source of error is due to possible incorrect labeling in the matching algorithm. This in turn implies inconsistent data to the GLSDC algorithm. This 'labeling error' is responsible for the large spikes for the estimations in scenarios *C*, *D*, and *E*. Furthermore, in the event that labeling errors occur in a continuous sequence of frames, the problem could lead to closed-loop instability.

The third source of error is associated with incorrect CG extraction procedure and it is mainly responsible of the high frequency chattering in the $|e_{MV}|$ signal.

Of particular interest in Fig. 8 is the region (on the right) associated with the condition $RB^m < 5$ m. Since the simulation is performed for 300 s., the condition $RB^m < 5$ m describes essentially the condition when the UAV is in the 3D refueling window under the tanker. In that particular condition the loss of visibility of multiple markers

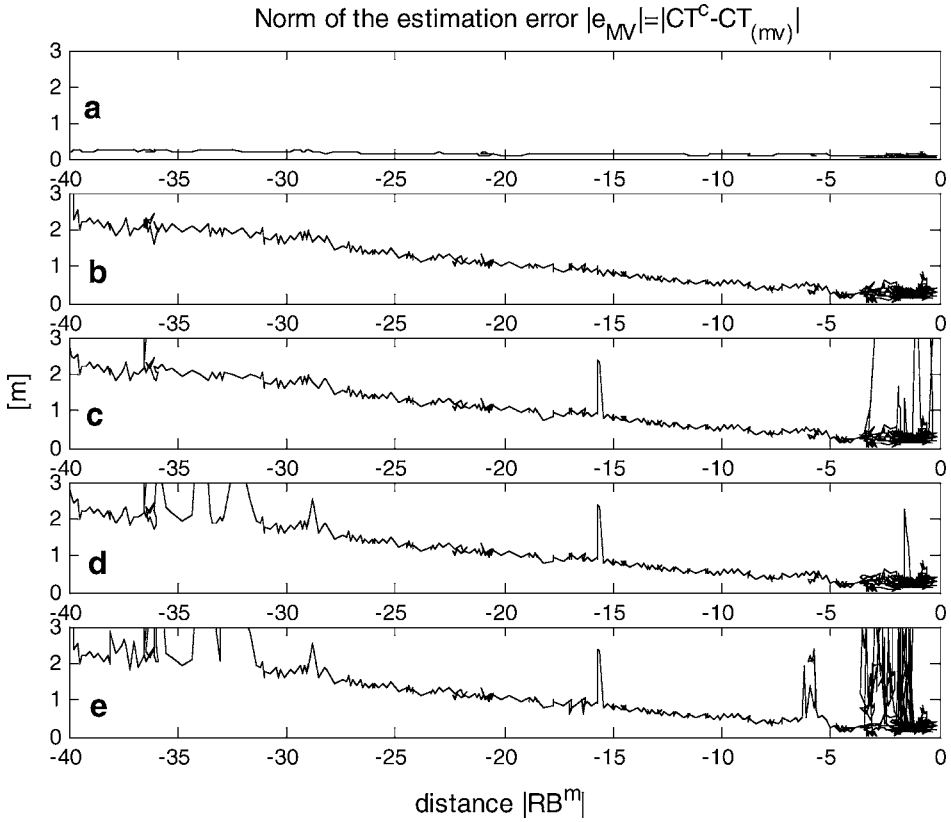

Fig. 8 $|e_{MV}|$ error as function of the distance RB^m.

is possible. This would then lead to deterioration of the performance of the matching algorithm causing the spikes associated with the scenarios C, D, and E.

This issue is even clearer from an analysis of Fig. 9 where $|e_{MV}|$ is shown as function of time (in the [35–85] s. window). This specific time interval is particularly relevant because it includes the final phase of the docking when the UAV enters and remains in the 3D refueling window below the tanker. Up to approximately 45 s (that is, when the UAV is still approaching to the tanker) a decrease in the $|e_{MV}|$ is noted for all the scenarios, with the exception of the 'ideal' and 'perfect' MV-based algorithms in Scenario A. Starting at approx. 45 s the error has a tendency to converge toward certain bounds. However, the scenarios C, D, and E present a large number of spikes, implying a lack of robustness for the associated labeling algorithms. Only scenario B shows reduced level of error throughout the simulation due to the desirable performance of the matching algorithm.

D. Influence of the Variable Number of the Visible and Labeled Markers

As previously described, the loss of visibility of one or more markers by the UAV camera might occur during the docking. If this occurs, the modified GLSDC algorithm has shown capabilities for adapting itself and continuing the estimation with the subset of the labeled markers available at that particular time step. Figure 10 shows the time histories of the number of the labeled markers, which can be up to the number of visible markers, used by the MV for the 5 scenarios for the same time interval as in Fig. 9. For the 'ideal' case of scenario *A* all the markers in the camera field of view are considered visible and correctly labeled. Starting from the value '$m_L = 9$', the reduction of the number of labeled markers is only due to loss of visibility due to physical occlusions (such as interference of the refueling boom).

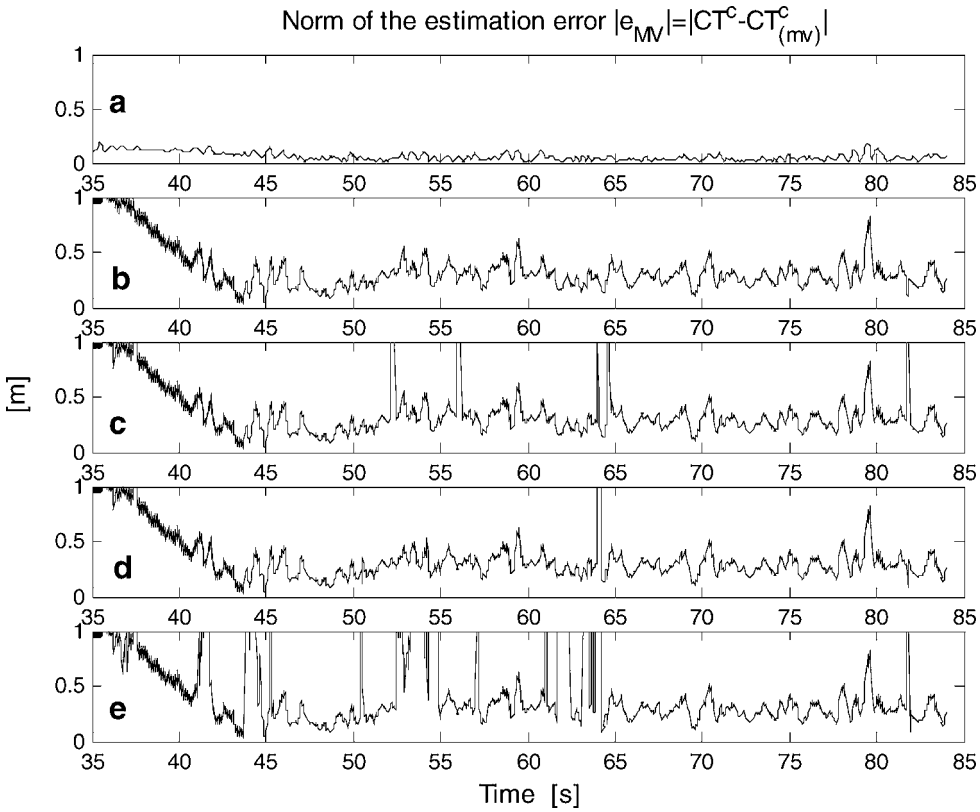

Fig. 9 $|e_{MV}|$ error as function of time.

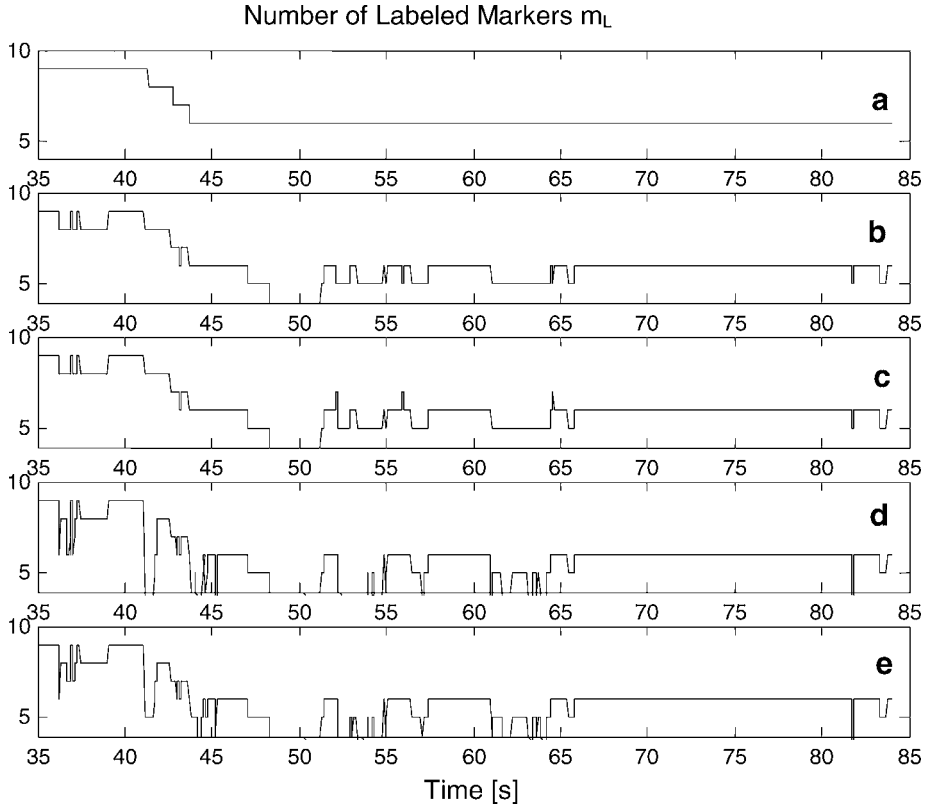

Fig. 10 Number of labeled markers in the interval [35–85] s (used by the DGLS algorithm).

For the scenarios B and C the performance of the labeling algorithms in terms of m_L seem to be equivalent. A close cross-analysis of Figs 9 and 10 reveal that the differences between scenario B and C in terms of m_L (Fig. 10) are responsible for the spikes in the estimation error in case C (Fig. 9).

The performance of the labeling algorithms in terms of m_L seem to be substantially deteriorated for scenarios D and E; even in this case it can be seen that the 'drops' in the value of m_L shown in Fig. 10 are associated with the spikes of the error shown in Fig. 9.

E. Quantitative Estimation of Performance

A quantitative analysis was performed separately for two specific phases within the AAR maneuver, that is for the docking maneuver (the first 45 s.) and for the time when the UAV is located within the 3D refueling window below the tanker (between 45 and 300 s.) associated with the condition $RB^m < 5$ m. The need to separate the analysis is due to the fact that in these two phases the MV estimation accuracy is clearly different.

For each of the 5 scenarios described above the performance have been compared in terms of the statistics of the measurement error $|e_{MV}|$. Thus, the mean, the Standard Deviation (SD), the maximum value and the mean number of labeled markers were evaluated. Table 2 shows the results for all the scenarios with scenario B being considered the 'baseline' scenario. As stated above, scenario A is just an ideal condition. From the analysis of the results in Table 2, a significant difference is evident between the performance of the 'baseline' scenario B and the performance of the scenarios C, D,and E,especially for the phase when the UAV has reached the 3D refueling window. Particularly, the performance degradation is clearer in terms of the standard deviation and maximum value of the error. In terms of physical units, for the scenario B the mean value of $|e_{MV}|$ is approx. 18 cm with a STD less than 6 cm and with a peak less than 35 cm.

Table 2 Overall Estimation performance of the 5 MV system.

	DOCKING Performance: $e_{MV} = CT^c - CT^c_{(mv)}$ [m]				3DW Performance: $e_{MV} = CT^c - CT^c_{(mv)}$ [m]			
	mean(e_{MV})	*std*(e_{MV})	max($\lvert e_{MV}\rvert$)	*Nmark*	*mean*(e_{MV})	*std*(e_{MV})	max($\lvert e_{MV}\rvert$)	*Nmark*
A	−0.084 (11%)	0.049 (6%)	0.200 (8%)	7.974 (107%)	0.003 (2%)	0.010 (20%)	0.032 (9%)	6.00 (106%)
B	**−0.709 (100%)**	**0.741 (100%)**	**2.479 (100%)**	**7.466 (100%)**	**0.181 (100%)**	**0.051 (100%)**	**0.341 (100%)**	**5.68 (100%)**
C	−0.677 (95%)	0.733 (99%)	2.621 (105%)	7.466 (100%)	0.168 (92%)	0.397 (794%)	5.090 (1492%)	5.70 (100%)
D	−0.644 (90%)	0.755 (102%)	3.466 (139%)	6.067 (81%)	0.176 (97%)	0.084 (168%)	1.054 (309%)	5.03 (89%)
E	−0.102 (14%)	1.411 (190%)	7.788 (314%)	6.728 (90%)	0.191 (106%)	0.889 (1778%)	8.348 (2448%)	5.30 (93%)

Table 3 Closed-loop performance.

	DOCKING Performance: $e_{MV} = CT^c - CT^c_{(mv)}$ [m]				3DW Performance: $e_{MV} = CT^c - CT^c_{(mv)}$ [m]			
	mean(e_{MV})	*std*(e_{MV})	max($\lvert e_{MV}\rvert$)	*Nmark*	*mean*(e_{MV})	*std*(e_{MV})	max($\lvert e_{MV}\rvert$)	*Nmark*
B MV in the Loop	−0.6491 (91%)	0.7380 (99%)	2.5558 (103%)	7.5383 (116%)	0.17947 (99%)	0.04973 (96%)	0.33073 (97%)	5.8563 (103%)
B no MV in the Loop	−0.7090 (100%)	0.7410 (100%)	2.4790 (100%)	7.466 (100%)	0.1810 (100%)	0.0510 (100%)	0.3410 (100%)	5.6880 (100%)
D MV in the Loop	−0.6053 (85%)	1.0006 (34%)	7.8360 (318%)	7.8360 (104%)	0.18961 (104%)	0.062856 (123%)	0.39673 (116%)	5.6378 (99%)
D no MV in the Loop	−0.6440 (90%)	0.7550 (102%)	3.4660 (139%)	6.0670 (81%)	0.1760 (97%)	0.0840 (168%)	1.0540 (309%)	5.0310 (89%)

F. Closed Loop Performance

In the final simulation the estimates from the MV-based algorithms were provided as feedback to a set of LQR-based docking control laws for final evaluation of the closed-loop performance [8]. Given the unsatisfactory performances of the algorithms in the scenarios C and E, the analysis was limited to the scenarios B and D. Although the MV estimation mechanism is clearly not dependent on the signals employed by the feedback controller, it is nevertheless of particular interest the comparison of the performance achieved when the MV is not in the feedback loop and the controller is driven by 'nominal' noise-free measurements (Table 2), and the performance achieved by using the MV-based measurements of the *CT* vector in the feedback loop. The results are shown in Table 3. An analysis of this table reveals that for scenario B the performance were fairly comparable for both cases. Some limited but significant differences between the two cases exist for scenario D. The origin of these differences is traced back to the spikes in the estimation error signal shown in Figure 7.

Another important conclusion from the analysis of the results in Table 3 is that both matching algorithms in the scenarios B and D provide *closed-loop* robustness to temporary loss of visibility of the markers by the UAV camera.

VIII. Conclusions

This paper describes the comparison between different feature matching algorithms within a simulation environment specifically built for the study of the MV-based Autonomous Aerial Refueling problem. In this study the attention was mainly focused on the performance, accuracy and robustness of the whole MV scheme when featuring different feature matching techniques. This analysis reveled a very high sensitivity of the pose estimation algorithm on possible errors in the feature matching algorithms. It was also shown that, assuming that the matching algorithms perform nominally, the overall closed-loop performance do not seem to be very sensitive to temporary/complete loss of visibility for some of the physical markers, provided a sufficient redundancy in their number. Furthermore the MV accuracy in the estimation of distance vectors seems to be appropriate given the size of a typical 3D refueling window. Finally, the proposed matching algorithm, which is based on a rigid translation model and on the measure of matching support (derived by the overall mean matching error between the two matching sets) has shown desirable robustness properties from extensive simulations at open-loop and closed-loop conditions.

References

[1]Korbly, R., and Sensong, L., "Relative Attitudes for Automatic Docking," *AIAA Journal of Guidance Control and Dynamics*, Vol. 6, No. 3, 1983, pp. 213–215.

[2]Kimmett, J., Valasek, J., and Junkins, J. L., "Autonomous Aerial Refueling Utilizing a Vision Based Navigation System," *Proceedings of the 2002 AIAA GNC Conference*, Paper 2002-4469, Monterey, CA, 2002, CD-ROM.

[3]Fravolini, M. L., Ficola, A., Napolitano, M. R., Campa, G., and Perhinschi, M. G., "Development of Modelling and Control Tools for Aerial Refuelling for UAVs", *Proceedings of the 2003 AIAA GNC Conference*, Paper 2003-5798, Austin, TX, 2003, CD-ROM.

[4]Pollini, L., Campa, G., Giulietti, F., and Innocenti, M., "Virtual Simulation Set-up for UAVs Aerial Refueling," *Proceedings of the 2003 AIAA Conference on Mmodeling and Simulation Technologies and Exhibits*, Paper 2003-5682, Austin, TX, 2003, CD-ROM.

[5]Philip, N. K., and Ananthasayanam, M. R., "Relative Position and Attitude Estimation and Control Schemes for the Final Phase of an Autonomous Docking Mission of Spacecraft," *Acta Astronautica*, Vol. 52, 2003, pp. 511–522.
doi: 10.1016/S0094-5765(02)00125-X

[6]Sinopoli, B., Micheli, M., Donato G., and Koo, T. J., "Vision Based Navigation for an Unmanned Aerial Vehicle", *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, South Korea, 2001, Vol. 2, pp. 1757–1764.

[7]Lee, C. H., and Joshi, A., "Correspondence Problem in Image Sequence Analysis", *Pattern Recognition*, Vol. 26, No. 1, 1993, pp. 47–61.

[8]Fravolini, M. L., Ficola, A., Campa, G., Napolitano, M. R., and Seanor, B., "Modeling and Control Issues for Autonomous Aerial Refueling for UAVs Using a Probe-Drogue Refueling System," *Journal of Aerospace Science Technology*, Vol. 8, No. 7, 2004, pp. 611–618.

[9]Addington, G. A., and Myatt, J. H., "Control-Surface Deflection Effects on the Innovative Control Effectors (ICE 101) Design,"Air Force Report, AFRL-VA-WP-TR-2000-3027, June 2000

[10]Air Force Link, online database: *http://www.af.mil/factsheets*

[11]Stevens, B. L., and Lewis, F. L., *"Aircraft Control and Simulation,"* John Wiley & Sons, New York, 1987.

[12]. Spong, M. W., and Vidyasagar, M., "*Robot Dynamics and Control,"* John Wiley, New York, 1989.

[13]Roskam, J. "*Airplane Flight Dynamics and Automatic Flight Controls – Part II*", DARC Corporation, Lawrence, KS, 1994.

[14]Blake, W., and Gingras, D. R., "Comparison of Predicted and Measured Formation Flight Interference Effects", *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, AIAA Paper 2001-4136, Montreal, Canada, 2001, CD-ROM.

[15]Gingras, D. R., Player, J. L., and Blake, W., "Static and Dynamic Wind Tunnel Testing of Air Vehicles in Close Proximity", *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, AIAA Paper 2001-4137, Montreal, Canada, 2001, CD-ROM.

[16]Hutchinson, S., Hager, G., and Corke, P., "A Tutorial on Visual Servo Control", *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 5, 1996, pp. 651–670.

[17]Umeyama, S. "Parameterized Point Pattern Matching and its Application to Recognition of Object Families," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 2, 1993, pp. 136–144

[18]Kim, W. Y., and Kak, A. C., "3-D Object Recognition Using Bipartite Matching Embedded in Discrete Relaxation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 3, 1991, pp. 224–251.

[19]Pla, F., and Marchant, J. A., "Matching Feature Points in Image Sequences through a Region-Based Method," *Computer Vision and Image Understanding*, Vol. 66, No. 3, 1997, pp. 271–285.

[20]Ferruz, J., and Ollero, A., "Real-Time Feature Matching in Image Sequences for Non-Structured Environments. Applications to Vehicle Guidance," *Journal of Intelligent and Robotic Systems,* Vol. 28, No. 1–2, 2000, pp. 85-123.

[21]Borrow, H. G., Ambler, A. P., and Burstall, R. M., "Some Techniques for Recognizing Structures in Pictures," *Frontiers of Pattern Recognitions,* Academic Press, New York, 1972, pp. 1–29.

[22]Rosenfeld, A., Himmel, R. A., and Zucker, S. W., "Scene Labelling by Relaxation Operators", *IEEE Transactions on Systems Man*, *Cybernetic*, Vol. 6, 1976, pp. 420–433.

[23]Strickland, R. N., and Mao, Z., "Computing Correspondence in a Sequence of Non Rigid Objectives," *Pattern Recognition,* Vol. 25, No. 9, 1992, pp. 901–912.

[24]Zhang Z., "On Local Matching of Free Form Curves", *Proceedings of British Machine Vision Conf.*, 1992, pp. 345–356.

[25]Kumar, S., Sallam, M., and Goldgof, D.,"Matching Point Features Under Small Nonrigid Motion," *Pattern Recognition*, Vol. 34, 2001, pp. 2353–2365.
doi: 10.1016/S0031-3203(00)00166-7

[26]Zhang, Z., Deriche, R., Faugeras, O., and Luong, Q. T., "A Robust Technique for Matching Two Uncelebrated Images Through the Recovery of the Unknown Equipolar Geometry", *Artificial Intelligence*, Vol. 78, 1995, pp. 87–119.
doi: 10.1016/0004-3702(95)00022-4

[27]Harlalick, R. M., Joo, H., Lee, C., Zhuang, X., Vaidya, V. G., and Kim, M. B., "Pose Estimation from Corresponding Point Data", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 6, 1989, pp. 1426–1446.

[28]Wilson, W., "Visual Servo Control of Robots Using Kalman Filter Estimates of Robot Pose Relative to Work-Pieces,Visual Servoing," *K. Hashimoto, ed., World Scientific*, 1994, pp. 71–104.

[29]Haralick, R. M., and Shapiro, L. G., "*Computer and Robot Vision*", Addison-Weley, Reading, Massachesets, 1993.

[30]Lowe, D. G., "Three-Dimensional Object Recognition from Single Two-Dimensional Image," *Artificial Intelligence*, Vol. 31, 1987, pp. 355–395.
doi: 10.1016/0004-3702(87)90070-1

[31]Lu, C. P., Hager, G. D., and Mjolsness, E., "Fast and Globally Convergent Pose Estimation from Video Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 6, 2000, pp. 610–622.

[32]Campa, G., Fravolini, M. L., Ficola, A., Napolitano, M. R., Seanor, B., and Perhinschi, M., "Autonomous Aerial Refueling for UAVs Using a Combined GPS-Machine Vision Guidance", *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA paper 2004-5350, Providence, Rhode Island, August 2004.

J. A. Mulder
Associate Editor