Vision Guided Manipulation for Planetary Robotics - Position Control

Kevin Nickels [a],*,

[a]*Department of Engineering Science, Trinity University, One Trinity Place, San Antonio, TX, USA, 78212-7200*

Matthew DiCicco [b], Max Bajracharya [b], Paul Backes [b]

[b]*Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr., Pasadena, CA 91109, USA*

Abstract

Manipulation systems for planetary exploration operate under severe restrictions. They need to integrate vision and manipulation to achieve the reliability, safety, and predictability required of expensive systems operating on remote planets. They also must operate on very modest hardware that is shared with many other systems, and must operate without human intervention.

Typically such systems employ calibrated stereo cameras and calibrated manipulators to achieve precision of the order of one centimeter with respect to instrument placement activities. This paper presents three complementary approaches to vision guided manipulation designed to robustly achieve high precision in manipulation. These approaches are described and compared, both in simulation and on hardware.

In-situ estimation and adaptation of the manipulator and/or camera models in these methods account for changes in the system configuration, thus ensuring consistent precision for the life of the mission. All three methods provide severalfold increases in accuracy of manipulator positioning over the standard flight approach.

Key words: vision guided manipulation, stereo imaging, planetary robotics, comparison study
PACS: 07.07.Tw, 42.30.Tz, 45.40.Ln

* tel.: +1 210 999 7543; fax: +1 210 999 8037
  Email address: `Kevin.Nickels@trinity.edu` (Kevin Nickels).

1 Introduction

Vision guided manipulation (VGM) systems designed for planetary exploration must address several concerns not necessarily shared by terrestrial systems. Some of these are concerned with the safety of the rover, leading to requirements such as ground (Earth-based) analysis of imagery from Hazard Avoidance Cameras (hazcams) before any manipulation [1]. Some concerns are environmental: the Martian surface, for example, sees daily thermal cycling from -100 to $+50^{o}C$, with significant amounts of small fine-grained dust. All components must be radiation-hardened, which severely restricts processor speed, memory capacity, and camera frame rate. Finally, any operational VGM system must be able to perform in these conditions with limited user interaction and without maintenance for the life of the mission. As such, the reliable performance of complex, high precision manipulation operations requires a hand-eye coordination strategy that accommodates these constraints.

This paper describes and compares three approaches to VGM that satisfy these constraints. Supporting evidence for the efficacy of these systems includes both simulation studies and implementation on a manipulator that has the kinematic and visual structure of the Mars Exploration Rover (MER) vehicles.

1.1 *History of Remote, Space-Based Manipulation Systems*

Manipulation systems for planetary exploration have evolved over the past 30 years from the early use of a telescoping sampling device on the Viking Landers in the 1970s to a single degree-of-freedom mechanism that was used to deploy the Alpha Proton X-Ray Spectrometer (APXS) from the Sojourner rover during the Mars Pathfinder mission in 1997. In an effort to increase the dexterity and available work volume for the placement of multiple science instruments, the Mars Polar Lander mission carried a four degree-of-freedom robot arm to be used for soil trenching and digging as well as placement of the Robotic Arm Camera (RAC) [2].

Currently, the Mars Exploration Rover (MER) vehicles carry a five degree-of-freedom robot arm (known as the Instrument Deployment Device, or IDD) that is used to place three in-situ instruments (the APXS, a Mössbauer spectrometer, and a microscopic imager) as well as place and hold a Rock Abrasion Tool (RAT) in order to abrade the weathered surface of a rock [1].

Both the most recent lander, Phoenix (2007), and the upcoming Mars Science Laboratory rover (MSL, launch in 2011) carry robot arms, and will utilize them to do in-situ science of increasing complexity. Pheonix utilized the robotic arm to scrape samples from the Martian surface and transport

them to the on-board Thermal and Evolved Gas Analyzer system and to the Microscopy, Electrochemistry and Conductivity Analyzer instruments for detailed chemical and geological analysis [3]. MSL will study Martian soil and rocks in unprecedented detail, utilizing the robot arm to carry samples from the surface to on-board test chambers for chemical analysis [4].

With the increasing demand for a higher level of science return in future surface missions, lander and rover-mounted robotic arms must exhibit a higher level of performance over current capabilities. As the capabilities and dexterity of each instrument deployment approach have evolved and improved over the last 30 years, the requirements associated with instrument placement precision have also increased. As a recent example, the MER mission IDD requirements set includes a precision placement requirement of 1 cm in position and 10 degrees in orientation with respect to a science target when the IDD is deployed from a stationary rover base [1].

1.2   Flight State-of-the-Art: Stereo Triangulation / Calibration

The current state-of-the-art in manipulation for planetary exploration relies solely on the use of a well-calibrated system to achieve the required precision with respect to instrument placement activities [1]. Typically the manipulation process is separated into two independent steps: the determination of the target range using a stereo camera pair and the subsequent control of the manipulator to the measured 3D location. This process involves two separate calibrations. The first calibration step determines the intrinsic and extrinsic parameters of a camera model relating the mapping between the 2D image plane of each of the stereo cameras and a physical 3D "world" coordinate frame. The second step involves kinematic calibration that is concerned with the determination of an accurate relationship between the manipulator joint angles and the 3D location of points on the manipulator's end-effector by way of the manipulator kinematic model. This step is necessary due to differences between the geometric model of the manipulator and the actual dimensions as a result of manufacturing tolerances and deformations of various kinds such as link flexure under load. Included in this step is the transformation from the manipulator 3D base frame to the 3D "world" frame.

Briefly, the manipulation process works as follows [5]: Each camera model is generated by identifying known 3D locations on a calibration fixture. Details about the standard camera calibration process can be found in [6][7]. Using these models, the 3D range to an identified target then can be determined via stereo correlation and triangulation. From this 3D range information the joint rotations that position the manipulator at the desired location in 3D space are determined using the arm's inverse kinematics. In this way the vision and
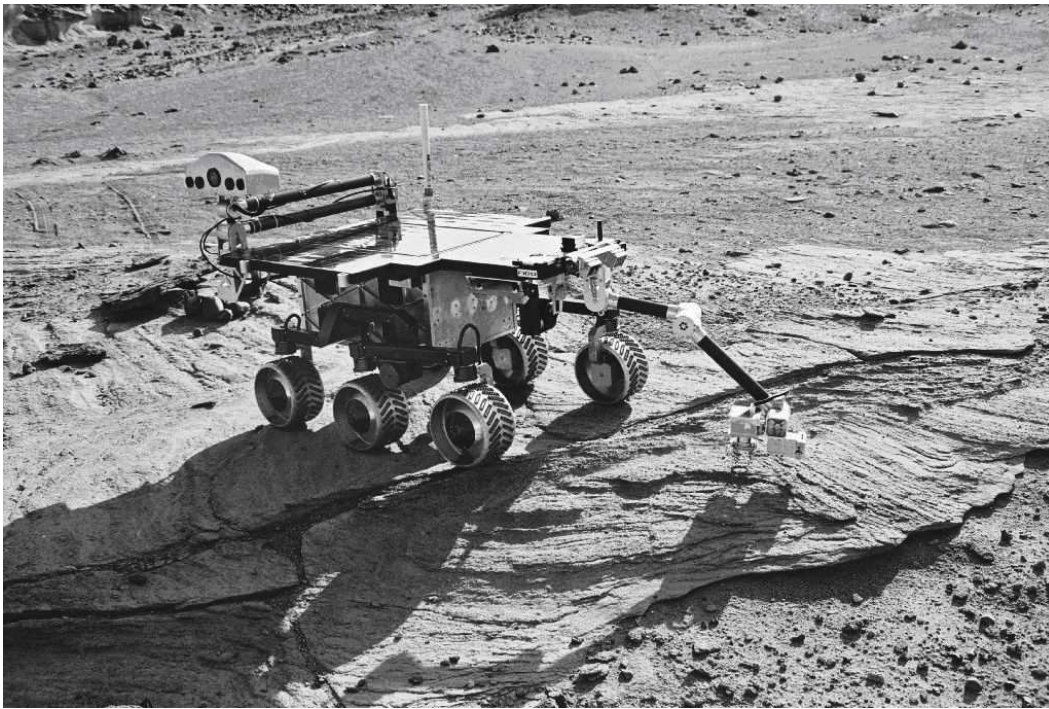
Fig. 1. The Field Integrated Design & Operations (FIDO) rover performs manipulator operations during a field trial designed to closely emulate Mars mission operation scenarios in Martian-like terrain.

manipulation steps are separated.

The difficulty associated with the standard approach is that sources of error tend to accumulate, ultimately reducing positioning precision, due to separation of the stereo vision and manipulator location processes. Sources of error in the manipulator location process include kinematic uncertainties in manipulator link lengths and reference frame transformations, unmodeled flexure of the manipulator links and base, and joint position uncertainties due to sensor resolution and backlash in the drive train. Additional errors involve imprecise stereo calibration and ranging accuracy. For example, with a baseline of 10 cm and a nominal target range of one meter, error analysis predicts that range accuracy errors alone can be as much as 6 mm [8].

Terrestrial operations such as the Field Integrated Design and Operations (FIDO) rover field trials (field tests designed to closely simulate Mars mission operation scenarios with a fully-instrumented rover in Martian-like terrain as shown in Fig. 1) have demonstrated the ability of such techniques to yield precision levels on the order of 1 cm [9]. However, on planetary missions, degradations in positioning precision may result from a decline in calibration fidelity due to changes in the system configuration as a result of environmental factors such as vibration during launch and landing, extreme thermal cycling, and inclement weather conditions. Indeed, Robinson et. al. [10] describe such a case of system degradation in the Mars Exploration Rover vehicles.

4

While the nominal level of precision is sufficient for current operations, future mission directives call for tasks of significantly increased complexity requiring millimeter level precision. In particular, the MSL Mars exploration mission proposes [4] a long-term, long-range rover with soil/rock sampling and precision placement of samples into a contained science laboratory for in-situ processing and analysis as a precursor to a planned Mars sample return.

1.3   Visual Servoing

In terrestrial robotics, the field of visual control of robot manipulators is dominated by visual servoing. Generally speaking, visual servoing specifies a desired configuration of the camera/manipulator system and places image measurements in a *control loop* to iteratively drive the error between the specified and measured configuration to zero over time.

A seminal review of Visual Servoing is given by Hashimoto [11], and a very accessible tutorial is given by Hutchinson et. al. [12]. The recent Springer Handbook of Robotics [13] includes a chapter describing the assumptions and methods behind visual servoing.

While the camera is often placed at the end-effector of the robot (termed the "eye-in-hand" configuration), fixed cameras are also used. The error term can be specified in image coordinates or in Cartesian coordinates. Finally, architectures exist for including the error term directly in the control loop, and yielding low-level control to the joint controllers.

The primary difference between the algorithms described in this paper and the various visual servoing architectures is the operational requirement that a human verify the proposed motions before the manipulator can be commanded. This requirement is discussed more in Section 3.1, and infers an increased emphasis on both precision in one control cycle and on fast convergence of iterative algorithms. Due to the emphasis of the space program on safety and verifiability, any algorithm that does not permit human verification of planned movements is unlikely to be deployed.

Under this requirement, each iteration of the control loop consumes one sol (planetary day) of rover time [1], and any iterative solution to VGM must provide significant benefits in accuracy with a single manipulator command.

---

[1] The delay is due to image downlink, science team review, and manipulation command uplink, not computation - none of the three algorithms described herein are computationally intensive for the rover.

2 Improved Vision Guided Manipulation

Each of the three VGM approaches described below attempts to model and reduce differences between the position of a given point on the end-effector of a manipulator as measured by the manipulator and its associated kinematic model and the position of that same point as measured by a pair of stereo cameras and their associated camera models.

They do this by modifying the nominal data flow in three different places. Figure 2 shows a block diagram of vision guided manipulation, and serves as a road map to how these algorithms modify the process.

A science target is selected manually (on Earth) by scientists, in a stereo range map. The 3D location of the target P_{tgt} cannot be directly accessed. Instead, the target is designated in the hazcam imagery as the two pixel positions $(u, v$ in the right and left image) to which the target projects. This vector [2] is designated as uv_{des}.

These pixels are triangulated utilizing Tranform T1 into the Camera Coordinate System (Camera CS) as P_{des}. Each of these techniques has been implemented utilizing both the eighteen-parameter CAHVOR model, a pin-hole camera model with radial distortion [6], and the twenty-one parameter refinement called CAHVORE, which adds a moving entrance pupil [7]. While a detailed explanation of this camera model is outside the scope of this paper, a quick summary of the parameters follows: $\mathbf{C}$ is the 3-D position of the focal center of the camera, $\mathbf{A}$ is the camera pointing vector, and $\mathbf{H}$ is the vector sum of $\mathbf{H}$, the sensor-plane horizontal vector, and the scalar multiplication of the image-coordinate u with the camera pointing vector, $\mathbf{A}$. Similarly, $\mathbf{V}$ is the vector sum of $\mathbf{V}$, the sensor-plane vertical vector and the scalar multiplication of the image-coordinate v with the camera pointing vector, $\mathbf{A}$, $\mathbf{O}$ is the optical axis, $\mathbf{R}$ is a 3-D vector representing the radial distortion, and $\mathbf{E}$ consists of polynomial coefficients that model the movement of the entrance pupil. The particular camera model used is not crucial to the techniques, but these have a long history of use at the Jet Propulsion Laboratory (JPL) and have well-understood performance characteristics and calibration and modeling tools associated with them. Without loss of generality, this paper exclusively refers to CAHVORE models below.

Transformation T2 is used to map this point to the Arm Coordinate System (Arm CS). This is a workspace transformation: normally a simple translation

---

[2] In some cases, the fiducial is spatially separate from the science target (i.e. 10cm normal to the rock at this location). The impact of this separation is not studied in this work. So here an ideal manipulation would have the fiducial located exactly where the science target is designated.
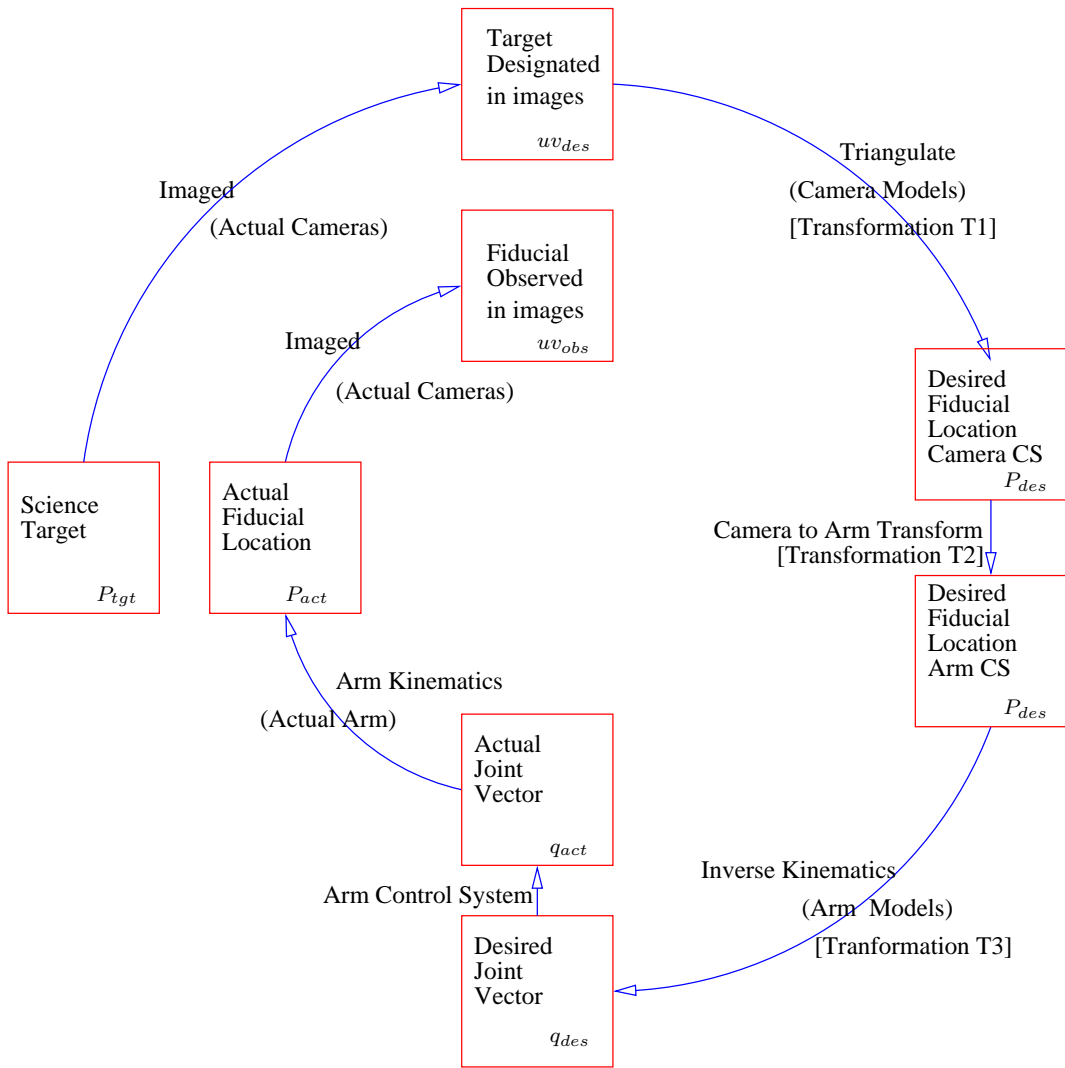
Fig. 2. Block Diagram of Vision Guided Manipulation.

modeling the mounting points of the camera and arm on the rover.

Inverse kinematics (Transformation T3) yield a desired joint configuration q_{des}. The Arm Control System then drives the arm to a new configuration q_{act}, which places the fiducial at a new place in the workspace P_{act}.

If all models were perfect, P_{act} would be identical to P_{tgt}. This new fiducial location can be observed in hazcam imagery at uv_{obs}, ideally identical to uv_{des}.

The image data are processed to extract the location of some reliable point on the manipulator that is referred to as a fiducial. While in the laboratory a simple fiducial such as shown in Figure 3 is used, any portion of the manipulator that can be reliably extracted from image data in uncontrolled lighting, such as the Mössbauer Contact Plate used in [14], can be used. Robustness to errors in feature localization were not studied in this work.
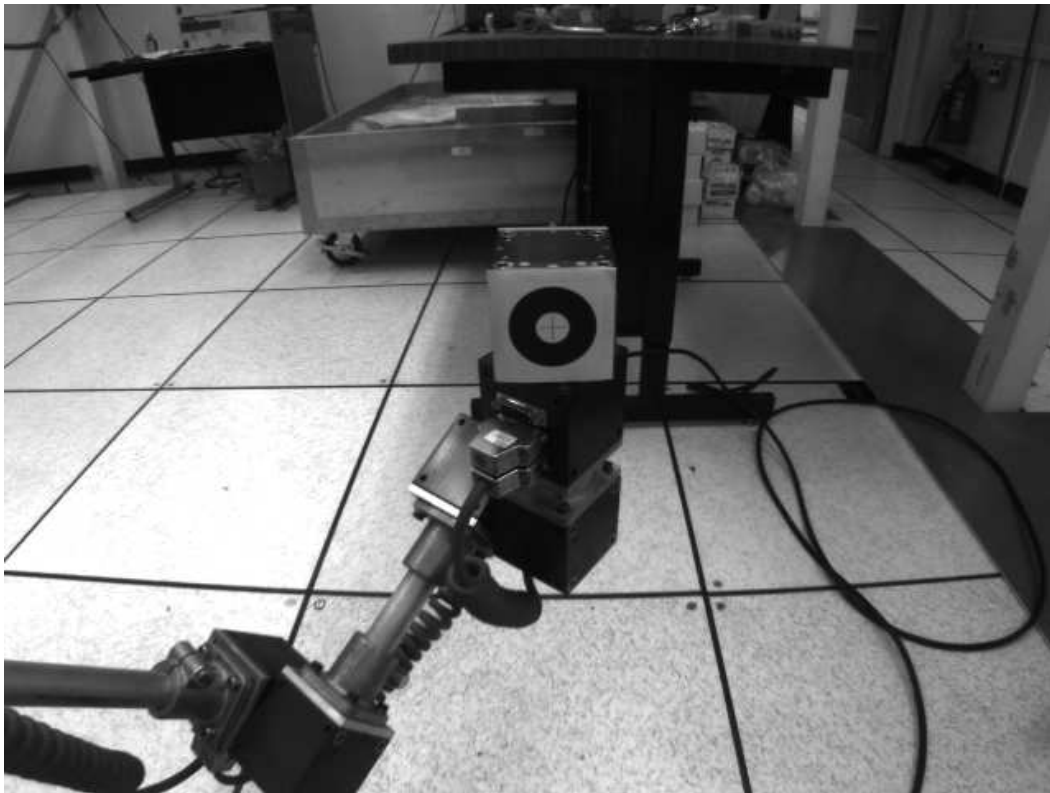
7

Fig. 3. Fiducial Marker Mounted on Manipulator. A fiducial marker mounted on the end-effector of the manipulator facilitates fast and reliable image processing.

2.1 HIPS

The Hybrid Image-Plane/Stereo Manipulation (HIPS) technique, proposed by Baumgartner et el. [15] and refined by Robinson et el. [10], generates camera models based on visual sensing of fiducial marker(s) on the manipulator's end-effector and the subsequent use of these models to position the manipulator at a target location specified in the image-plane of each camera of the stereo pair.

HIPS is most similar to an alternative technique for hand-eye coordination known as Camera-Space Manipulation (CSM) [16]. Both HIPS and CSM move the manipulator through a set of predetermined joint configurations and observe the image-plane locations of a distal fiducial. However, in the CSM approach the inverse problem is solved directly using a least-squares minimization to determine the joint rotations that will locate the end-effector at the desired pose in the image-plane of each participant camera. Ravela et. al. [17] similarly estimate a weak perspective affine transform between an object model and the observed object as the object is moved.

2.1.1 Static HIPS

For each camera the CAHVORE parameters are initially estimated using a predetermined set of typically twenty to fifty manipulator poses. This "preplanned trajectory" constitutes a broad sample of both the image-space of the vision sensor and the joint-space of the manipulator. At each pose a set of images from the camera pair and the manipulator joint angles are recorded.

The 2D image space location of the fiducial in each image is extracted through image processing. The 3D work space location of the fiducial is computed using the nominal forward kinematics of the robot. This set of 5D vectors (3D position and 2D image plane location) are used to compute a new set of CAHVORE parameters.

These camera models are referred to as "Static HIPS Models." They are used in place of the nominal camera models in Transformation T1 in Figure 2 when triangulating the designated target position in imagery (uv_{des}) to arrive at a camera-space position for the target (P_{des}). The remainder of the data flow remains the same.

2.1.2 On-Line HIPS

The aforementioned approach to manipulator control addresses the systematic errors that are present in the standard flight approach. These include separate camera calibrations, transformations between various frames, and manipulator kinematic errors such as link lengths and joint offsets.

However, stochastic errors that occur due to finite image-plane cue detection, camera modeling errors, and inaccurate knowledge of joint angles arising from sensor resolution, orientation-dependent droop, joint backlash, etc., are not necessarily accounted for in this approach.

One solution to this problem, proposed by Robinson [10] et. al., is to divide the trajectory to the target, or "transition" trajectory, into a series of intermediate steps. At each intermediate goal the camera models are updated by use of data from newly acquired images of the end-effector. This generates new HIPS camera models that are locally more precise than the "Static HIPS Models."

By use of the updated camera models, the target range is recomputed as before and then used to refine the necessary manipulator joint angles to position the end-effector at the target. As distance between the end-effector and the target decreases, the computed coordinates of the end-effector and target become affected by increasingly similar errors, so that the difference between their coordinates become increasingly small. Therefore precise placement of the manipulator's end-effector is achieved.

End-Effector Position Error Correction (EPEC) is a simple and a computationally efficient algorithm that has been developed and implemented for VGM by Bajraachayra et. al. [14]. The EPEC algorithm locates and triangulates the position of a single fiducial on the end-effector and servos the end-effector to a position designated in a stereo image pair. In doing so, the EPEC algorithm maintains an error correction vector between the fiducial's position as predicted by the forward kinematics of the arm and as found by the stereo camera pair. It then uses this error vector to compensate for kinematic position inaccuracy.

If the fiducial is occluded or imaging is no longer available, a previously calculated error vector can be used locally to reduce positioning error in the final placement. The algorithm requires that the stereo cameras be calibrated well enough to triangulate the position of a point found in the two cameras and that the arm be calibrated well enough to do roughly accurate small Cartesian motions. Weakly calibrated cameras would be sufficient since metric reconstruction is not necessary, however convergence will be significantly faster with more information. The transformation between the stereo camera frame and the arm base frame must be approximately known in order to facilitate predicting the fiducial position.

2.2.1 Static EPEC

Static EPEC analyzes all available end-effector locations to generate a table of correction vectors. These correction vectors, CV, are simply the difference between the kinematic prediction of the end-effector, P_{kin}, and the visually detected position of the end-effector, P_{vis}, and are indexed by the visually detected fiducial location. Thus, for a target visually designated in the camera images P_{vis_tgt}, the command sent to the manipulator P_{kin_tgt}, is:

$$CV(P_{vis}) = P_{kin} - P_{vis}, \qquad \forall (kin, vis) \in \text{training set}$$
$$P_{kin\_tgt} = P_{vis\_tgt} + CV(P_{vis\_tgt})$$

The EPEC correction is used in Transformation T2 in Figure 2, to transform a desired target location from the camera coordinate system to the manipulator coordinate system.

The *EPEC table* is the aggregation of the EPEC correction vectors. To use the EPEC table for VGM, a novel target is selected in the image planes, and nominal camera models are used to triangulate this target to a 3D location. The correction vector closest to the detected target location is located and added to the command vector before it is passed to inverse kinematics and

finally to the manipulator.

DiCicco et. al. have shown [18] that the ability of EPEC to correct for visual-kinematic mismatches is directly proportional to the workspace distance from the nearest correction vector. In addition to the use of the Correction Vectors for positioning, they also offer a valuable insight to the overall health of the rover's vision-manipulation system. Nickels et. el [19] describe their use in this light.

2.2.2 On-Line EPEC

Due to the direct correlation of EPECs precision and the distance to the nearest correction vector location, it is best to acquire samples as close in space to the desired target location. If the fiducial is visible near the target location, the most recent correction vector is taken instead of performing a table lookup as above. The original formulation of EPEC [20] exclusively uses the most recent correction factor, with no additional memory in the form of a lookup table.

If this process is repeated until convergence is achieved, the result would be termed a "position-based visual servo control" (PBVS) structure [12][13]. However, it is important to note that EPEC has been developed with the constraints of planetary robotics in mind, which leads to an emphasis on the amount of correction in the first iteration of the PBVS loop.

2.3 DHTune

DHTune is an algorithm originally developed at Johnson Space Center [21] to aid in calibrating Robonaut's seven degree-of-freedom arm to assist with autonomous tasks requiring improved hand-eye coordination.

DHTune operates in a manner similar to HIPS, in that the manipulator is driven through a set of predefined joint angles and a fiducial is observed at each pose and the realized joint angles are recorded. The fiducial locations as computed by the kinematics and as computed by the camera system differ by some amount.

Where the algorithms differ is that while HIPS computes new camera models (Transformation T1 in Figure 2) that minimize this difference, DHTune modifies the nominal kinematic parameters (Transformation T3 in Figure 2) to minimize this difference. If the joint angle offsets are the only parameters included in the optimization, DHTune is similar to Skarr's CSM method [16]. However, typically the twist and offset of some of the links are also included

in the optimization.

The use of the DHTuned kinematics mirrors that of HIPS closely. A target is designated in the image planes and nominal camera models are used to triangulate the 2D points to a 3D location. The modified kinematics are used to compute the inverse kinematics to move from this 3D location to a joint vector that is used to command the robot. This usage would, in the taxonomy utilized in this paper, be referred to as *static DHTune*.

A re-weighting scheme such as online HIPS (described in Section 2.1.2) would be expected to yield similar results, but this extension has not been performed or evaluated. Thus, *online DHTune* is not compared with the other algorithms below.

2.4   A Common Framework for the Approaches

When considering the data flow shown in Figure 2, the differences between the VGM algorithms can be summarized as follows. The baseline algorithm uses nominal camera and arm models, each optimally calibrated within their own systems. The system-level accuracy of the manipulator/camera system is characterized by the workspace residual $||P_{tgt} - P_{act}||$.

The HIPS manipulation technique generates new camera models to be used in place of the nominal camera models in the $uv_{des} \rightarrow P_{des}$ transformation (T1) to minimize this residual. Nominal arm models are used. The EPEC technique adds a workspace offset to P_{desc} to get P_{desm} (T2) before the inverse kinematics to minimize this residual. Nominal camera and arm models are used. Finally, the DHTune technique replaces the nominal arm model in the $P_{desm} \rightarrow q_{des}$ transformation (T3) with an new arm model computed to minimize the residual, while retaining the nominal camera models.

3   Experimental Comparisons

One objective of this study was to consider, on as even a basis as possible, the relative strengths and costs of the various methods for VGM. The amount of training data needed and the efficacy of both static and online versions are considered.

While quantitative comparisons of the computational complexity of each algorithm were not performed, they are all fast-converging or off-board computations that are well within operational requirements for space-relevant hardware [10] [14] [21].
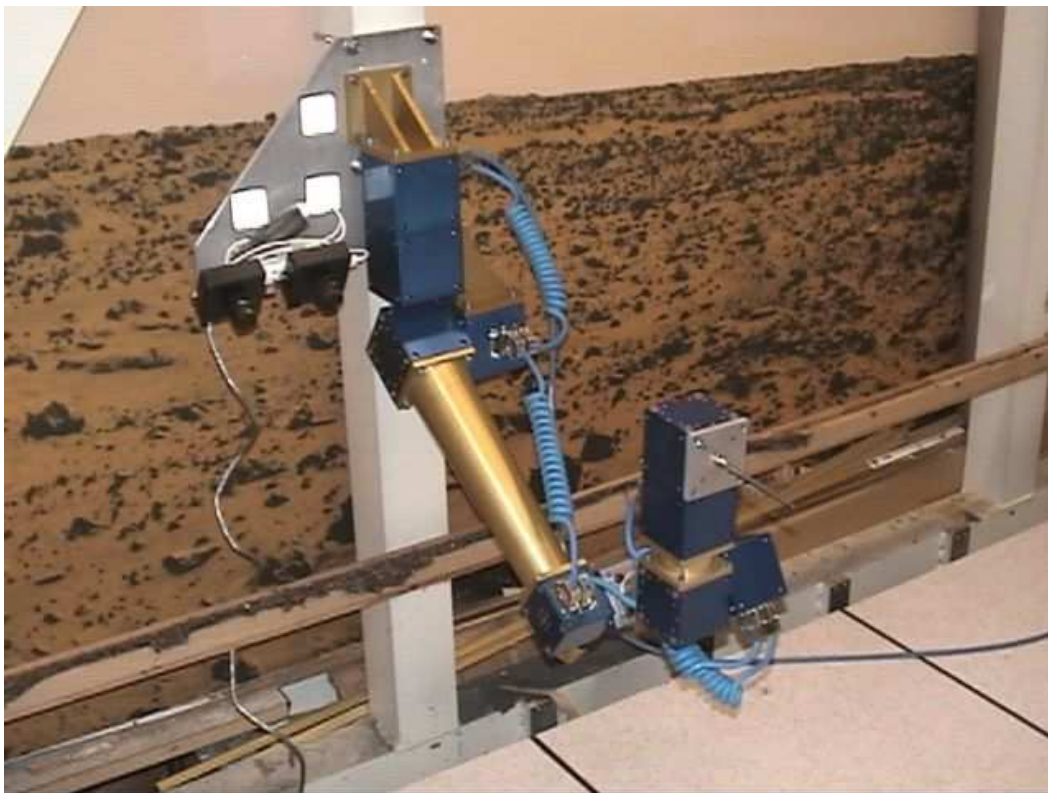
Fig. 4. The Five Degree of Freedom Modular manipulator

To this end, HIPS, EPEC, and DHTune were implemented and tested on a planetary manipulator mock-up at JPL. This five degree of freedom arm, shown in Figure 4, is one of the manipulators in JPLs Modular Robotic Testbed, where robotic arms of various kinematic configurations can be quickly and easily constructed from AMTEC PowerCubes and steel piping. This system has the approximate kinematic structure and camera locations of the Mars Exploration Rover (MER) vehicle [5],[8]. The camera configuration has two cameras with an 10 cm baseline pointing 30 degrees down. They have 2.8 mm lenses and a 640x480 CCD with $4.65 \mu m$ pixel size. At a target range of one meter, with careful calibration this configuration corresponds to 1 pixel being approximately 1 mm in lateral error and a 0.3 pixel error in stereo disparity matching error being 10 mm in range error.

3.1 Experimental Comparison of Static Algorithms

There are many reasons why the static type of VGM algorithm is useful for planetary robotics. First and foremost, the use of the low-level joint controllers and flight software to move the arm ensures that standard fault protection checks, such as collision with the rover, joint limits, etc., are performed before the motion is carried out. For example, Baumgartner et. al. [5] describe the

13

flight software for the Instrument Deployment Device (IDD), including fault protection checks, for the MER vehicles. The offboard nature of the corrections allows the current science target flow to be maintained, as described in [1]:

> When not in use during surface operations, the IDD is stowed in a position similar to the launch position. Stereo images obtained using the front hazcams show the IDD work volume and any science targets that may be present in it. Hazcam images are used on the ground to identify target surfaces within the work volume, and to determine their position and orientation relative to the rover. This position and orientation information is then used to generate command sequences that direct the IDD to position payload elements against target surfaces.

In addition, there is no potential for erroneous fiducial detection results or other dynamic phenomena to cause unexpected motions of the IDD. Given the harsh and uncontrolled lighting conditions, such as those shown in Figure 5, developing feature extraction algorithms that are robust is a challenge. With static VGM, rover planners can evaluate each potential move for safety and efficacy before uplinking the commands.

However, this type of control has some clear drawbacks, also described in [1]:

> This commanding approach is simple and robust, but it imposes important restrictions. No closed-loop positioning using Hazcam images onboard is performed, and no deployment of the IDD without analysis of Hazcam images on the ground is permitted.

This implies that the "one-step correction" methods such as the three considered below may be the best that can be reliably achieved within reasonable constraints on rover safety.

3.1.1   Data Collection

A set of 100 target points, arranged in a 3D grid in the overlapping portions of the stereo image space and the arm workspace, was chosen. The arm was commanded to these targets, and for each target, five joint angles q_{act} and two 2D image plane locations of the detected fiducial (four locations) uv_{act} were recorded. The nominal kinematic model and nominal camera models were also recorded.

These 100 data points are randomly split into two sets: 49 points for testing, and 50 points for training. One point was at an extremum of the workspace and was excluded. The 50 training points are randomly sub-sampled into additional training sets of 25, 10, 5, and 1 data points.
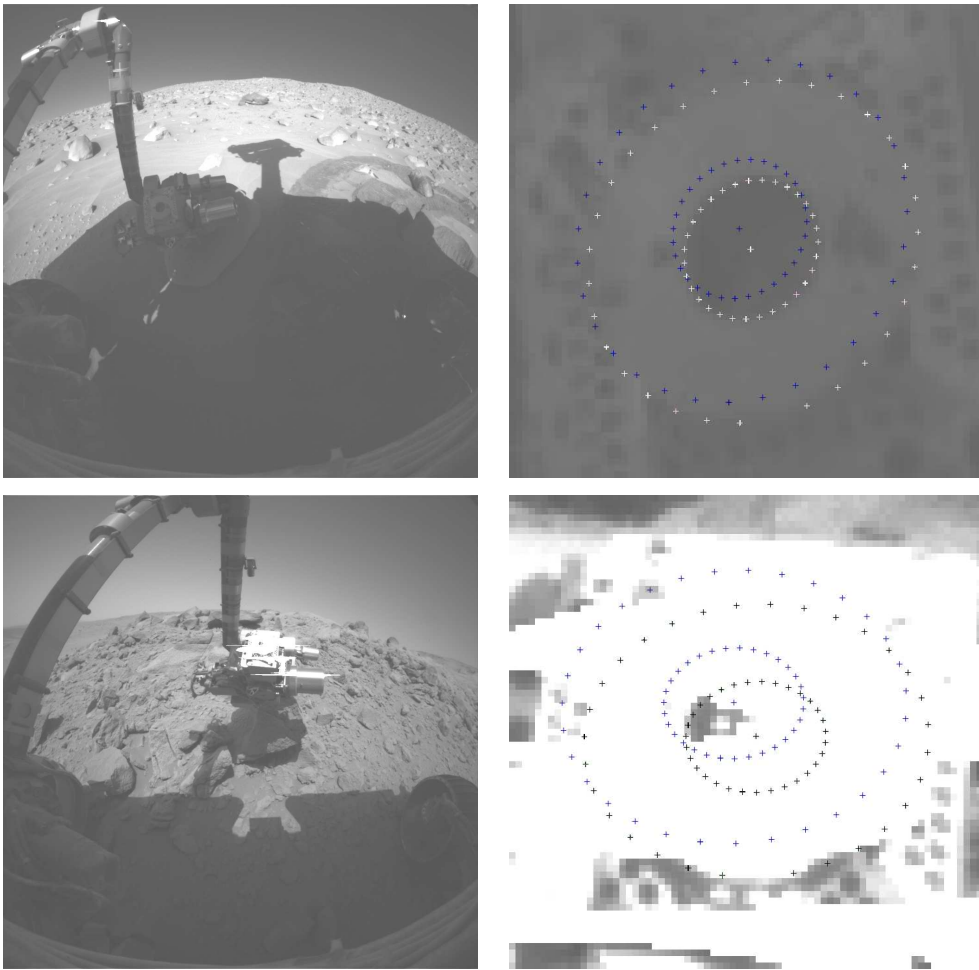
(a) Hazcam Images   (b) Zoomed-In on "Fiducial"

Fig. 5. Automatic feature extraction is a challenge for Planetary Robotics. This figure shows Two Sets of Images from the Spirit Mars Exploration Rover. In (a), the left Hazard Avoidance Camera Image is shown, and in (b), the area around the Mössbauer Contact Plate is shown. The blue crosses (in the web version of this article) represent the kinematic prediction of the Contact Plate, and the black or white crosses represent the output of a fiducial detector.

3.1.2   Evaluation of Algorithms

In this study, each algorithm was given the training data (1, 5, 10, 25, or 50 points) and the nominal models as described in Section 3.1.1. This enabled each algorithm to create an appropriate *model* of the arm/camera system to be used in the right side of Figure 2. For HIPS, this is a new set of camera models to be used in the $uv_{des} \rightarrow P_{desc}$ transformation (T1). For EPEC, this is a table of workspace correction vectors, to modify the Camera to Arm Transform (T2). For DHTune, this is a new arm model to be used in the $P_{desm} \rightarrow q_{des}$ transform (T3).

The VGM algorithms are then given the 49 recorded uv_{act} from the test data as described in Section 3.1.1 as targets, and used the new models to generate a new joint command, q_{vgm}. The recorded arm commands, q_{act}, known to have produced *exactly* the given target locations, were not available to the VGM algorithms.

Since it is difficult to compare two joint configurations directly, the nominal forward kinematics are used to project the two configurations into the manipulator coordinate system, where the workspace residual $||FwdKins(q_{vgm}) - FwdKins(q_{act})||$ is used to evaluate how well the VGM mapping generalizes to new configurations. This residual is referred to as the Workspace Residual below. The average absolute distance between the projections of $FwdKins(q_{vgm})$ and $FwdKins(q_{act})$ on the image planes is called the Camera Residual.

3.1.3 Results - Static Algorithms

All three VGM models yielded several-fold improvement over the flight-standard approach. As mentioned above, the VGM algorithms were given recorded target locations and the resulting joint vectors were compared to the recorded ones. All data are summarized in Figure 6, and results for the best case of 50 training points are given in Table 1.

The fiducial detector used in these experiments, described in [14], returns the location of the fiducial to the nearest tenth of a pixel. As mentioned above, 1 pixel at a range of 1m in these cameras yields around 1.3 mm of error.

3.2 On-Line

While the static algorithms test how precise a single joint command can be made, the online versions look at utilizing dynamic feedback from the cameras for improvement. Online versions of HIPS and EPEC have been developed,
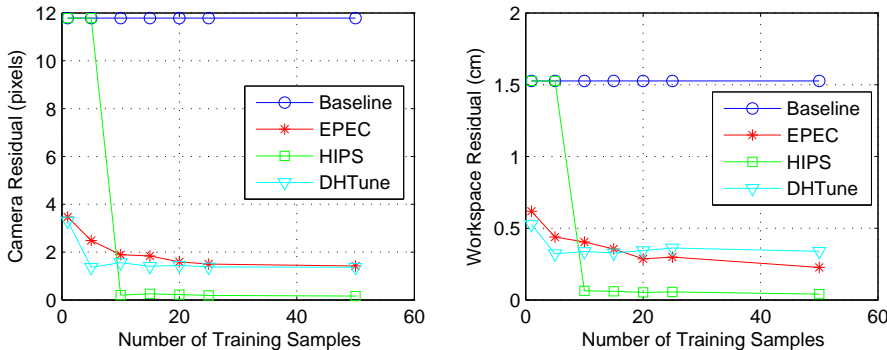

Fig. 6. Impact of Amount of Training Data on Arm-Camera Residuals.

Table 1
Results for 50 training points - Static

Approach	Residual Mean (mm)	Residual Std Dev (mm)
Flight Standard	15.26	2.55
HIPS	0.41	0.31
EPEC	2.27	1.11
DHTune	3.39	3.20

and are tested below. Because DHTune and HIPS operate similarly, and there are well-established and highly optimized methods for computing CAHVORE camera models, an online version of DHTune has not been pursued.

3.2.1 Mean Residual after Convergence

To test the online versions of the HIPS and EPEC algorithms, four representative targets are chosen from the 50 test points above. According to existent "best practices," the two approaches utilize slightly different methods of acquiring new samples to refine the arm/camera mapping.

For the HIPS approach, the workspace between a home "standoff" position and the target is segmented to provide additional visual/kinematic samples to the algorithm, and new camera models are generated as described above. As explained in Section 2.1.2, the computed 3D coordinates of the end-effector and target become more nearly affected by the same errors, so the differences between their coordinates become smaller. Therefore, the residual tracked is the root sum of squares distance between the predicted fiducial location and the detected fiducial location.

For the EPEC approach, the nearest correction vector to the specified target is utilized, and the arm commanded to move to this location. A new correction vector is generated at this location, and a (hopefully smaller) corrective move is issued. The residuals between the goal position and the detected position are recorded.

The online versions of both algorithms improve upon the static versions by another order of magnitude. As shown in Figure 7, the EPEC workspace residual improves from a mean for the four targets of 4.59 mm to a mean of 0.23 mm. For HIPS, the final workspace residual improves from a mean of 0.77 mm to a mean of 0.06 mm. Step 1 illustrates the starting point for the on-line versions, the input from the "static" approach described above, Step 2 is the first on-line corrected position of the manipulator.
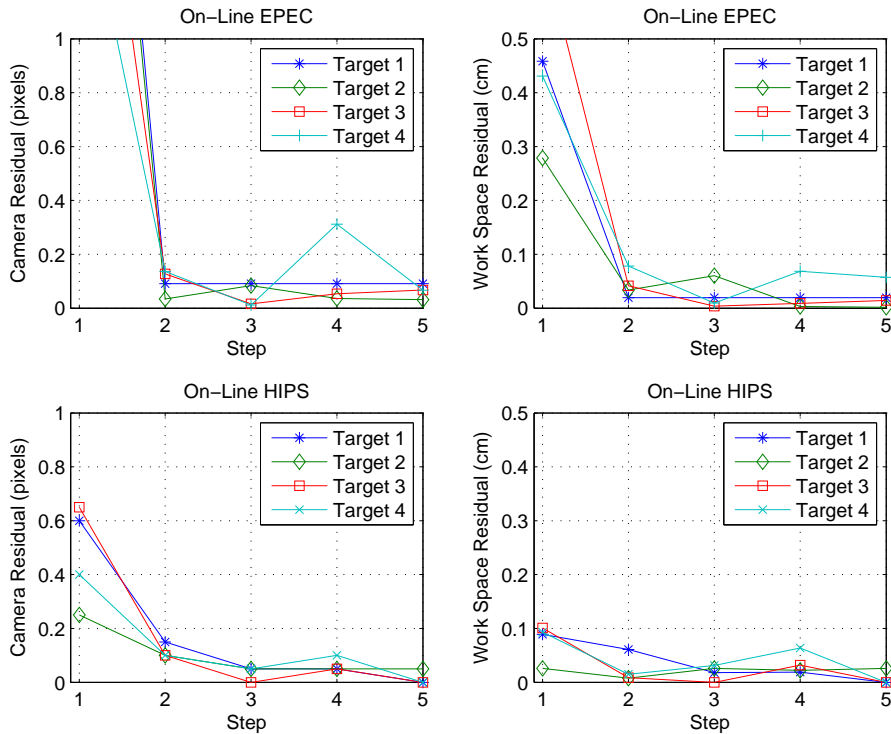
Fig. 7. Residuals vs Iteration for Online VGM for four representative targets. The 50 point training set was used for these runs.

It should be noted that the trajectory generator for this arm has occasional errors of up to 0.3 degrees, leading to workspace control errors of up to 1.7 mm, depending on the configuration. This can be clearly seen, for example, in the increased residuals in step 4 of one run in the EPEC data of Figure 7. This particularly affects small workspace moves, possibly limiting the workspace convergence of the EPEC algorithm. However, the mean level of convergence is at the limit of the fiducial detection so this effect cannot be evaluated.

Since the original targets for this data set were specified in the workspace, the residuals used for the EPEC algorithm are limited in convergence by the fidelity of the nominal camera models, which are used to project these targets to the image plane. In particular, unmodeled misalignments of the optical axes could never be corrected, and the computed image-plane targets could never be achieved. This is a modeling error, not an error in control, but it does reflect in the residuals above. Therefore, only convergence toward the target locations in the horizontal directions, which can always be achieved, is reflected in the residuals shown in Figures 7 and 8.
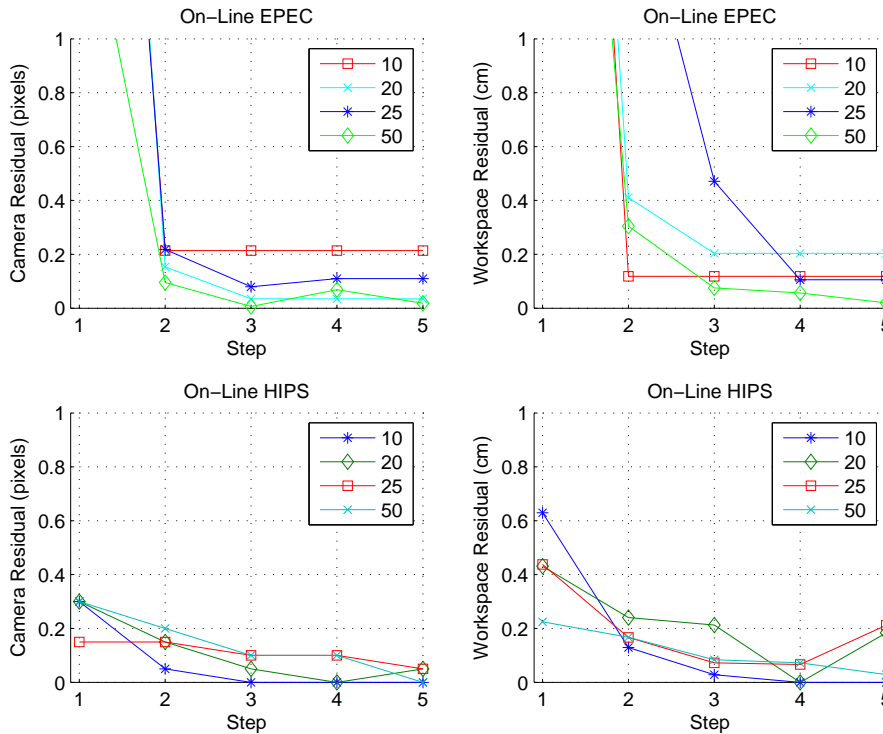
18

Fig. 8. Residuals vs Iteration for Online VGM for various training sets. The number in the legend indicates the size of the training set used to initialize the online algorithm.

3.2.2 Impact of Static Training Set Size

To investigate the impact of the size of the static training set on convergence of the algorithms, the test described in Section 3.2.1 was repeated for each training set. Figure 8 illustrates the results from a representative run. If multiple online samples are available, both EPEC and HIPS converge to their final residual, down to the limit of the fiducial detection, with little variation due to initial training set size. Only the starting residual is affected.

4   Conclusions

This paper described several approaches to the control of manipulators using estimation, stereo correlation and triangulation for space exploration applications. HIPS generates new camera models to minimize the 3D residual between the arm and camera systems, while DHTune generates a new arm model to do the same job. In contrast, EPEC adds a workspace correction vector to the visual results to arrive at a new arm command.

The three algorithms all have very low online cost since they simply modify existing model parameters (HIPS/DHTune) or add a offset (EPEC), and are applicable to any situation where a rover's computational constraints are high and positional improvement in one itration is an important metric.

The results of comparison of the "static" versions of these algorithms (where only one arm command is allowed) with differing training set sizes were presented, with DHTune improving the mean residual for a test set of 50 poses from 15.25 mm for the Flight Standard approach to 2.27 mm, EPEC improving the residual to 2.27 mm, and HIPS improving the residual to 0.41 mm, all for 50 training points. For this situation, if at least ten arm/camera samples can be acquired, HIPS provides the best correction factor for vision guided manipulation, approximately an order of magnitude over the flight-standard approach.

The results of online versions of two of the algorithms were presented, which drive the image-plane residual to sub-millimeter accuracy (the limit of the fiducial detector used) in one to two iterations. Each algorithm can work in concert with any of the "static" algorithms, and require only one extra arm/camera sample and arm command per iteration. Results were presented with several representative targets as well as different starting positions. In these circumstances, EPEC reduced the residuals to an average of 0.23mm, and HIPS to an average of 0.06mm, over the four targets. Since both algorithms converge well under the resolution limit of the fiducial detector used, EPEC's computational simplicity becomes more attractive in this situation.

In conclusion, if at least ten arm/camera samples can be acquired, HIPS will provide the best correction factor for vision guided manipulation, approximately an order of magnitude over the flight-standard approach. HIPS was developed to provide this type of correction [15], and these experiments appear to validate its utility here. If on-line corrections are feasible, either HIPS or EPEC can drive the positioning error to the limit of a 0.1 pixel fiducial detector in one or two additional iterations. While HIPS was modified [10] to utilize online correction factors, EPEC was originally designed [14] as an online correction algorithm.

5   Future Work

The final result of VGM is to position an instrument on a planetary surface, not to position a fiducial in a specified position. To investigate the impact of the VGM algorithms on the positioning of instruments remote from the fiducial, the next step in this study will be to include the *orientation* of the manipulator in the study.

It may also make sense to collect correction vectors at a given location over time as the rover performs its duties, then apply some combination of them to correct future moves in this kinematic-visual location. The optimal combination of vectors has yet to be investigated, however.

In addition, it would be useful to the design of the Mars Science Laboratory manipulator arm if these results could be repeated on an arm with kinematics relevant to that arm, which will likely be longer and more flexible than the MER IDD. This makes vision guided manipulation a much more attractive notion for future missions.

Acknowledgment

References

[1] S. Squyres, et al., Athena Mars rover science investigation, J. Geophysical Research 108 (E12) (2003) 8062.

[2] H. U. Keller, et al., The MVACS robotic arm camera, J. Geophysical Research 106 (E8) (2001) 17,609–17,622.

[3] P. Smith, The Phoenix mission to Mars, in: Proc. IEEE Aerosp. Conf., Vol. 1, 2004, p. 342.

[4] D. Savage, G. Cook-Anderson, NASA selects investigations for the Mars Science Laboratory, Jet Propulsion Laboratory, Press Release 04-398 (Dec. 2004).

[5] E. T. Baumgartner, R. G. Bonitz, J. P. Melko, L. R. Shiraishi, P. C. Leger, A. Trebi-Ollennu, Mobile manipulation for the Mars Exploration Rover, IEEE Robot. Automat. Mag. 13 (2) (2006) 27–36.

[6] D. Gennery, Least-squares camera calibration including lens distortion and automatic edition of calibration points, in: A. Grun, T. Huang (Eds.), Calibration and Orientation of Cameras in Computer Vision, Springer-Verlag, 2001, pp. 123–136.

[7] D. B. Gennery, Generalized camera calibration including fish-eye lenses, Intl. J. of Computer Vision 68 (3) (2006) 239–266.

[8] J. N. Maki, et al., Mars Exploration Rover engineering cameras, Journal of Geophysical Research 108 (E12) (2003) 8071.

[9] E. Tunstel, et al., FIDO rover field trials as rehearsal for the NASA 2003 Mars Exploration Rovers Mission, in: Proc. of 9th Intl. Symp. on Robotics & Applications, 5th World Automation Congress, Orlando, FL, 2002, pp. 320–327.

[10] M. Robinson, E. Baumgartner, K. Nickels, T. Litwin, Hybrid Image Plane/Stereo (HIPS) Manipulation for robotic space applications, Autonomous Robots 23 (2) (2007) 83–96.

[11] K. Hashimoto (Ed.), Visual Servoing, World Scientific, 1993.

[12] S. Hutchinson, G. Hager, P. Corke, A tutorial on visual servo control, IEEE Trans. Robot. Automat. 12 (5) (1996) 651–670.

[13] F. Chaumette, S. Hutchinson, Visual servoing and visual tracking, in: Springer Handbook of Robotics, 2008, pp. 563–583.

[14] M. Bajracharya, M. DiCicco, P. Backes, K. Nickels, Visual end-effector position error compensation for planetary robotics, J. Field Robot. 24 (5) (2007) 399–420.

[15] E. T. Baumgartner, P. C. Leger, P. S. Schenker, T. L. Huntsberger, Sensor-fused navigation and manipulation from a planetary rover, in: Proc. SPIE Sym. on Sensor Fusion and Decentralized Cntrl. in Rob. Sys., Boston, 1998.

[16] S. B. Skaar, W. H. Brockman, R. Hanson, Camera space manipulation, Intl. J. Robot. Res. 6 (4) (1987) 20–32.

[17] S. S. Ravela, T. J. Schnackertz, Jr., R. A. Grupen, R. S. Weiss, E. M. Riseman, A. R. Hanson, Temporal registration for assembly, in: In IROS Computer Vision Workshop, 1995.

[18] M. DiCicco, M. Bajracharya, K. Nickels, P. Backes, End-effector pose error compensation – error analysis, in: Proc. IEEE Aerosp. Conf., Big Sky, MT, 2007.

[19] K. Nickels, M. Bajracharya, A. Trebi-Ollennu, R. Liebersbach, Kinematic-vision residuals analysis, in: Proc. IEEE Aerosp. Conf., Big Sky, MT, 2007.

[20] M. Bajracharya, M. DiCicco, P. Backes, Vision-based end-effector position error compensation, in: Proc. IEEE Aerosp. Conf., Big Sky, MT, 2006.

[21] K. Nickels, E. Huber, M. DiCicco, Hand-eye calibration using active vision, in: Proc. IEEE Aerosp. Conf., Big Sky, MT, 2007.