10th ESA Workshop on
Advanced Space Technologies for
Robotics and Automation

'ASTRA 2008'

11-13 November 2008 at ESTEC,
Noordwijk, the Netherlands

REALISTIC IMAGE GENERATION FOR TESTING VISION-BASED AUTONOMOUS RENDEZVOUS

M. McCrum[1], S. Parkes[1], M. Dunstan[1], I. Martin[1]

[1]Space Technology Centre, University of Dundee, Dundee, Scotland, DD1 4HN, United Kingdom.
markmccrum@computing.dundee.ac.uk

ABSTRACT

This paper describes the development of a tool for generating realistic synthetic images of planetary rovers and planet surfaces for the purpose of testing vision-based autonomy algorithms. Such algorithms have been used on the NASA Mars rovers and will be used heavily on ExoMars for navigation. Computer simulation is a useful complement to testing in artificial physical test beds and natural terrestrial analogue environments, as it offers advantages of flexibility, availability, repeatability, speed and low cost. Computer generated images may also be useful in other areas such as operator training tools, supporting mission operations, and public engagement. The aim of this work is to create a generic, reusable visualisation tool, not tightly coupled to any single rover design or simulation environment, which can generate images in close to real time. This will enable researchers to perform testing in a more realistic environment than would otherwise be practical, without the need to develop their own image generation software. The work is an extension of the Planet and Asteroid Natural Scene Generation Utility (PANGU) developed by the University of Dundee.

VISION-BASED ROVER AUTONOMY

Lunokhod 1 was the first mission to provide mobility on the surface of a planetary body. The relatively short radio time delay between Earth and the Moon, where Lunokhods 1 and 2 operated, permitted direct interactive control of the rovers from the ground. Even at this range, control was made difficult by the radio time delay. The next planetary rover, NASA's Sojourner, operated on Mars, where round-trip radio delays on the order of 40 minutes made interactive control impossible. Thus Sojourner, like the Viking landers before it, carried out a daily command sequence prepared by mission controllers. It also included a degree of autonomy, allowing it to plan and follow a safe path to a destination. NASA's 2003 Mars Exploration Rovers expanded on this level of autonomy, greatly boosting the science return of the missions. The next generation of rovers; Mars Science Laboratory and ExoMars will take this capability even further in order to make the maximum use of the time available without requiring detailed supervision from the ground.

Computer vision plays a key role in rover autonomy. Important examples of this are path planning and visual odometry. In path planning, stereo vision is used to construct a 3D model of the terrain around the rover. A path planning algorithm may then be used to determine a safe path for the rover to reach a waypoint designated by mission controllers. This process may be repeated at intervals during a traverse to allow the path to be refined. Visual odometry is the use of computer vision techniques such as feature tracking or optical flow to determine the distance and direction in which the rover has moved during a traverse. This can greatly increase the accuracy of a traverse by enabling the rover to compensate for wheel slip and slide. Both of these techniques raise the level of abstraction of commands which can be given to the rover, to something which more closely matches the goal of the operator, and enable the rover to cope with an uncertain environment. Potential future vision-based autonomy capabilities include and the automatic identification of scientific targets of opportunity; and Single Cycle Instrument Placement (SCIP), which will allow the rover, on a single command, to traverse to a designated rock, and place an instrument on it [1].

TESTING VISION-BASED ALGORITMS

To be able to test and validate vision-based autonomy algorithms it is necessary to generate input images as test data. During the initial development of an algorithm, the volume of test data required may be small and the required fidelity low. However, as the system matures it will become increasingly important to have test images which are as representative as possible, and the amount of testing will increase as the limits of the system are explored, the performance characterized and parameters tuned. Eventually it will be necessary to test the algorithms in closed loop with other rover systems. One approach to testing is to use a rover prototype or test bed. This may be done in synthetic environments such as the Mars Yard or in a Mars Analogue environment such as the Atacama Desert. While this kind of testing is essential, it is too expensive and time consuming to be used exclusively. It is also only relatively late on in the development of a rover that a prototype will become available, and then only to a select group of developers.

A complimentary approach is to use computer graphics techniques to create synthetic images simulating the view from rover cameras. The many benefits of this approach include flexibility, availability, repeatability, speed and cost. However, autonomy developers are unlikely to want to spend time creating realistic image generation software. It is suggested therefore that there is a need for a software tool which meets the following criteria:

1. Generates realistic images from a rover point of view including the rover environment and any portions of the rover in the camera field of view.
2. Allows visualisation of a rover in its environment from a third person perspective.
3. Can be made available to the rover community.
4. Has a simple, effective interface to allow easy integration with whatever simulation and testing environment a developer is using.

The next section examines relevant existing tools in this field in relation to the above criteria.

RELATED WORK

Computer simulation in general plays an important role in the development, validation and operation of planetary rovers, with a wide variety of tools having been developed. Of these however, only a few are intended to be used for testing vision-based algorithms.

ROAMS

NASA's ROAMS tool [2] is a comprehensive planetary rover simulation environment incorporating models of rover systems and the planetary environment, which can be used to close control loops for testing rover software. It includes the ability to generate synthetic camera images and to visualise simulations, and therefore meets the first two criteria. However it is an internal tool, not easily accessible outside NASA, so failing to meet the third criteria. There is little published information regarding external interfaces.

Mission Simulation Facility (MSF)

The NASA Mission Simulation Facility [3], was developed to provide a realistic test environment for autonomy developers. Its focus is to provide a framework to allow distributed simulations of rovers to be built by integrating models of rover systems and the environment. It also includes a set of representative models of rover systems. MSF is freely available to users outside NASA; however it is hampered by its use of the complex DMSO High Level Architecture, and in particular by its reliance on a Runtime Infrastructure implementation which is no longer freely available. This makes it difficult in practice for outside developers to make use of it. Although MSF can be integrated with NASA's Viz visualisation system to provide high quality graphics for camera input generation or simulation visualisation, the Viz system itself is not available outside NASA. In its current freely available version, MSF meets the second and third criteria.

CNES Rover Simulator

The initial development of navigation algorithms for ExoMars has been carried out using a simulator developed by CNES [4]. There is little published information about this tool, which does not appear to be widely available. However, the published images lack some realism and it would appear that the image generator is tightly coupled with other aspects of the simulation system, making it difficult to use in other contexts. It therefore meets only the second of the criteria.

Discussion

None of the tools described meet all four of the criteria established. They either do not offer sufficient realism, or they are tightly coupled within a particular simulation environment, or they are not available to users outside the originating institution. The remainder of this paper will describe work carried out by the University of Dundee to enhance the PANGU tool to meet these criteria.

PANGU

The Planet and Asteroid Natural Scene Generation Utility (PANGU) [5–10] was originally developed to generate realistic test data for vision-based guidance systems for planetary landers, and has been widely used in that context. It consists of a suite of tools to allow the production of realistic synthetic models of planet and asteroid surfaces, and a viewer program to generate simulated sensor images of these surfaces in near real time. As well as cameras, LIDAR and RADAR sensors are supported. PANGU has also been extended to allow spacecraft models to be visualised in the context of in-orbit rendezvous simulation as part of the ESA HARVD activity [11]. Since PANGU already provides the ability to render images of planetary environments and spacecraft, and has been used successfully by a wide range of groups, it provides a useful basis in terms of planet surface simulation, vehicle simulation, visualisation and interfaces on which to build a rover simulation capability.

The PANGU rover enhancement work carried out had two aims. The first was to identify a comprehensive set of requirements for a tool to generate synthetic camera images for rover simulation, and for simulation visualisation. To accomplish this, a survey of the rover simulation literature was conducted and inputs obtained from potential end users. The planetary science literature was also consulted to determine what surface features are important at rover scale, with emphasis on Mars. The second aim was to demonstrate the potential of the PANGU tool to meet these requirements, by implementing a subset of them and creating a demonstration simulation as a proof of feasibility. The features implemented as part of the feasibility study are described below.

NEW FEATURES

Rover Model Support

The capability to load and visualise spacecraft models has been extended to allow a rover model to be loaded into the tool. This model includes the geometry of the rover, and its mechanical structure, indicating the position and type of the rover joints and the position and orientation of any on-board sensors. The inclusion of a geometric model permits a third person view of the rover, useful for visualising the results of simulations, and also allows any portions of the rover which are in the field of view of a rover camera to be displayed. Furthermore, it is necessary to allow shadows cast by the rover onto the terrain to be rendered. Incorporating a representation of the mechanical structure of the rover means that the current rover configuration can be conveniently communicated to PANGU simply in terms of joint angles. Additionally, the model allows the position of rover mounted cameras to be tracked. This frees a client application from having to track the position of cameras itself.

To prepare a rover model for PANGU, an XML file describing the rover structure is produced. This describes each component of the rover, the joints which link the components and the position of any cameras. In addition, for each part of the rover structure such as a wheel or a rocker, a 3D model is prepared. This may be in PANGU’s internal format or in the international standard VRML97 format. To simplify the process of preparing a rover model, we have also created plug-ins for the Blender 3D modelling tool [12], which can provide a graphical method of describing the rover structure. The plug-ins allow each part to be exported as a VRML file and the XML description to be generated automatically (Fig. 1). This approach has been used successfully to prepare a PANGU rover model based on a detailed CAD model of the ExoMars rover. The model was exported from the CATIA CAD package in the STEP format. A third party tool was used to prepare this model and translate it into the VRML format. Finally, Blender was used to build the XML description of the rover structure.

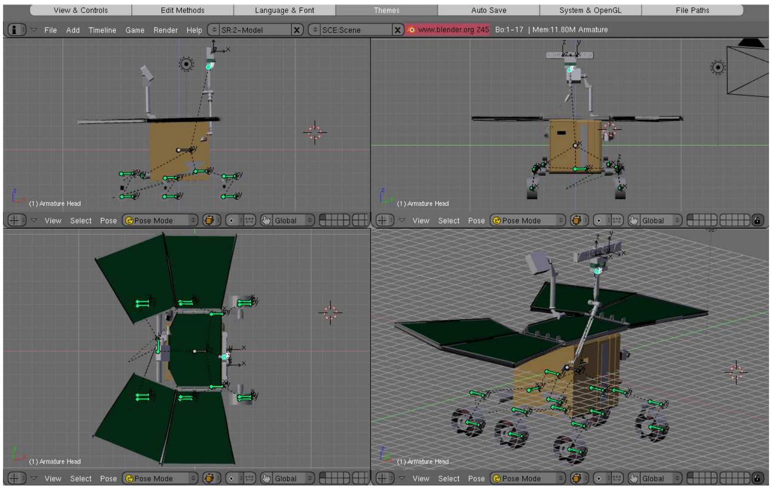

Fig. 1. Rover model preparation in Blender.

TCP Network Interface

It is not intended to develop PANGU into a complete rover simulator. Rather, the role of PANGU is to provide input for sensors such as cameras, and to allow simulations to be visualized. Therefore it is important that PANGU has a simple and effective interface to other simulation tools. This is achieved through a TCP interface, with PANGU acting as a server. In order to support rover applications, the new rover functionality has been exposed through this interface. Through it, client applications may query and set the configuration of rover joints, set the rover position and attitude and retrieve images from the rover cameras. To support simulation of rover interaction with the terrain, the elevation of the surface may be queried and entire patches of the terrain may be retrieved as a digital elevation model. Using a simple TCP interface provides a great deal of flexibility. For example, client applications have been written using Java, C, and MatLab. The simple client-server architecture also allows rudimentary coarse-grained parallelism to be achieved by using a separate PANGU server for each rover camera.

DEMONSTRATION SYSTEM

To demonstrate the new functionality in PANGU, two models of the ExoMars rover were created, and two demonstration applications developed. The first model was created from a detailed ExoMars CAD model. The second was created from scratch in the Blender 3D modelling software by an undergraduate student working from ESA publicity images of the ExoMars Phase B rover study. Of the two demonstration applications, the first is written in JAVA and simply allows the rover joints to be directly manipulated and displays the views from rover cameras (Fig. 2). The second is written in MatLab and provides a simple kinematic model of the rover, to allow rover movement over a surface to be simulated. The MatLab model is able to interface to the PANGU server using the same Java classes as the first application.

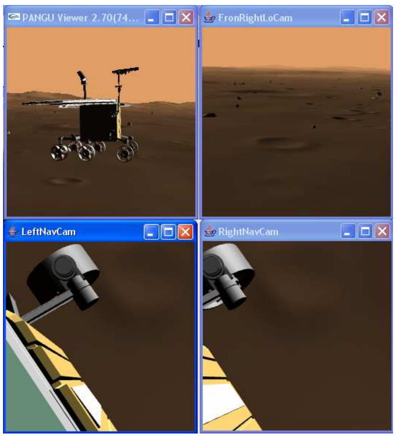

Fig. 2. Demonstration program showing views from rover cameras.

WORK IN PROGRESS

The enhancements so far made to PANGU demonstrate its potential for rover simulation. However, there remains a great deal of scope for further development. Features currently under development or being considered for the future are described below:

Enhanced shadows

Shadowing computation is a difficult problem in computer graphics. Currently no generally applicable solution is available. Rather there are a multitude of techniques offering different trade-offs in terms of accuracy, speed, memory requirements, supported light types and constraints on scene geometry. PANGU employs a number of different techniques, each optimized for a specific situation: shadowing calculations for the terrain model are performed offline and stored as a shadow map. This is necessary because surface models are potentially very large and complex. Self shadowing of spacecraft and rovers is done dynamically, since they will move, and is accomplished through the depth-buffer shadow mapping technique. Enhancements to this shadowing system will allow the rover to cast shadows onto the terrain and the terrain on to the rover. For efficiency reasons, the Sun is treated as a point light source during shadow calculation. The resulting shadows are hard, with no penumbrae. While this approximation is often reasonable, it becomes less acceptable at low sun angles, where shadows will tend to have large penumbrae. For terrain shadowing, this may be compensated for by combining multiple shadow maps for point lights which sample the surface of the sun. For real-time shadow generation however, this technique is far too slow. Various techniques to simulate penumbrae in real-time exist, however most have the aim of creating shadows which are perceptually correct, with the emphasis on looking nice rather than being realistic.

Improved boulder models

Boulders in PANGU were developed to appear realistic when viewed from above, and usually from a significant altitude. From a rover perspective they appear far less realistic. A more capable boulder system is currently under development. This will consist of a tool which can generate more realistic boulder models to form a library of boulders. The user will also have more control over boulder distributions.

Rover tracks

Rover tracks are often quite prominent in images from the Mars Exploration Rovers. The ability for a rover to create tracks on a PANGU surface model will be useful both in enhancing the realism of the generated images and also in providing a useful visual cue to users during simulation visualisations.

Visualisation Support

PANGU's current graphics facilities are aimed purely at providing realistic images of the rover in its environment. When used as a visualisation tool however, it would be useful to have the capability to allow the representation of additional abstract data in the scene. Examples of this include:

- Superimposing navigation maps onto the terrain.
- Showing waypoint markers.
- Showing the field of view of a camera.
- Altering the colour of rover wheels depending on whether or not they are in contact with the ground.

OTHER APPLICATIONS

The current focus of PANGU development for rovers is for use in rover testing. In addition to this, a number of other interesting applications can be foreseen:

Training

The benefits of simulations to training and education are well established. PANGU's capabilities would form a good basis to construct training simulation software [13]. Such a package would give rover operators the opportunity to familiarize themselves with rover operations. They might also be useful for early operational readiness testing and also for 'virtual field trials' [14]. Simulators could also be very useful for members of the rover science team. Early access to such a tool would be a good way of communicating the capabilities and limitations of the rover prior to the mission, and may help to streamline procedures and inform ‘concepts of operation’.

Public Engagement

PANGU could also be a useful tool for public engagement, both through the creation of educational simulations and games, and the production of high quality images (Fig. 3) and videos for the press.

CONCLUSION

The PANGU tool for generating test data for vision-based lander GNC algorithms has been enhanced to facilitate its use in rover simulations. The tool permits an articulated rover model to be loaded and visualised in a realistic Mars or Moon like environment. External tools may use PANGU as a visualisation server through a TCP interface. This interface permits control of the rover configuration, generation of images either from a third person view or from a rover-mounted camera, and queries of the terrain as a digital elevation model. The feasibility of using these new features for rover simulation has been demonstrated. Work is ongoing to expand the capabilities of PANGU in this area.

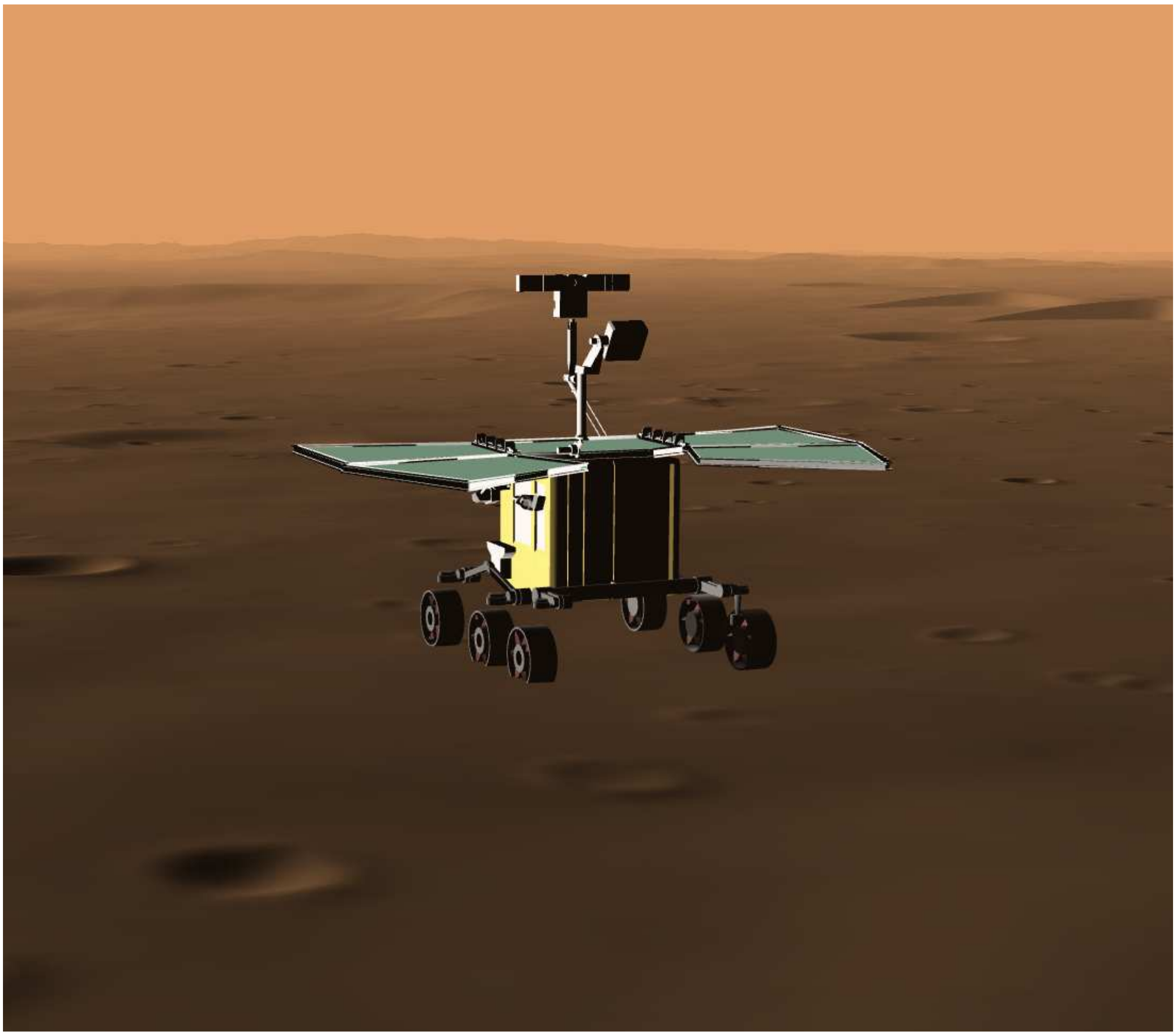

Fig. 3. PANGU Visualisation of ExoMars surface operation.

ACKNOWLEDGEMENTS

Thanks to Ishan Sardana for creating the ExoMars rover model seen in the figures.

REFERENCES

[1] T. Estlin, L. Pederson, and B. Williams, "Demonstrating Robotic Autonomy in NASA's Intelligent Systems Project", *8th ESA Workshop on Advanced Space Technologies for Robotics and Automation*, 2004, ESTEC, Noordwijk, the Netherlands.

[2] A. Jain, et al, "Recent developments in the ROAMS planetary rover simulation environment", *IEEE 2004 Aerospace Conference*, 2004, Big Sky, Montana, USA.

[3] G. Pisanich, L. Plice, C. Neukom, L. Flückiger, M. Wagner, "Mission Simulation Facility: Simulation Support for Autonomy Development", *42nd AIAA Aerospace Conference*, 2004, Reno, USA.

[4] L. Rastel, and M. Maurette, "Autonomous Navigation: A Development Roadmap for Exomars", *9th ESA Workshop on Advanced Space Technologies for Robotics and Automation*, 2006, ESTEC, Noordwijk, the Netherlands.

[5] S. Parkes, I. Martin, and S. Strandmoe, "Planet Surface Simulation for BEPI Colombo", *DASIA 2001 Data Systems in Aerospace*, 2001, Nice, France.

[6] S. Parkes, I. Martin, M. Dunstan, and S. Mills, "Mercury surface simulation for Bepi Colombo Lander", *DASIA 2002 Data Systems in Aerospace*, 2002, Dublin, Ireland.

[7] S.M. Parkes, I. Martin, and I. Milne. "Lunar surface simulation - modelling for vision-guided lunar landers". *DASIA'99 Data Systems in Aerospace*, 1999, Lisbon, Portugal.

[8] S. Parkes, I. Martin, and M. Dunstan, "Planet surface simulation with PANGU", *Eighth International Conference on Space Operations*, 2004, Montreal, Canada.

[9] S. Parkes, M. Dunstan, I. Martin, and S. Mancuso, "Radar altimeter simulation with PANGU", *DASIA 2005 Data Systems in Aerospace*, 2005, Edinburgh, Scotland..

[10] S. Parkes, M. Dunstan, I. Martin, P. Mendham, and S. Mancuso, "Planet surface simulation for testing vision-based autonomous planetary landers", *57th International Astronautical Congress*, 2006, Valencia, Spain.

[11] M. McCrum, M. Dunstan, and S. Parkes, "Realistic Image Generation for testing Vision Based Autonomous Rendezvous", *GNC 2008: 7th International ESA Conference on Guidance, Navigation and Control Systems*, 2008, Tralee, Ireland.

[12] www.blender.org

[13] S.M. Parkes, M. Dunstan, and I. Martin. "Preparing mission operators for lunar and mars exploration". In *Space Operations Mission Management, Technologies and Current Applications.* L. Bruca, J.P. Douglas and T. Sorensen, Eds. American Institute of Aeronautics and Astronautics.

[14] J.S. Norris, M.W. Powell, J.M. Fox, K.J. Rabe, I-Hsiang Shu. "Science Operations Interfaces for Mars Surface Exploration", *2005 IEEE International Conference on Systems, Man and Cybernetics*, 2005, Hawaii, USA.