Postprint: Implementation and Application of a Lunar Surface Close-Range Detection Simulation System

Authors: Zhang Jing, Jianjun Liu, Gao Xingye, Chen Zheng

Date: 2017-09-26T00:00:00+00:00

Abstract

A lunar surface close-range detection simulation system was constructed for real-time lunar surface scene display and payload motion simulation. The system, based on the open-source Object-oriented Graphics Rendering Engine (OGRE), combined with modeling software 3DMAX, integrated development tool Visual Studio, 3DS-to-MESH format conversion tool 3ds2mesh, and terrain generation plugins, completed the construction of three-dimensional solid models of the lander and related payloads, and achieved realistic simulation of the lunar surface environment. A parameter panel was designed using the open-source CEGUI graphics interface library for data display and human-computer interaction. Additionally, a three-dimensional scene debugging interface was built based on the Microsoft Foundation Class library to facilitate direct control of three-dimensional scene updates and adjustment of entity model poses, simulating the payload detection process. Finally, simulation experiments were conducted using a panoramic camera as an example.

Full Text

Implementation and Application of a Lunar Close-Range Exploration Simulation System

Zhang Jing[1,2,3], **Liu Jianjun**[1,3], **Gao Xingye**[1], **Chen Zheng**[4]

[1]National Astronomical Observatories, Chinese Academy of Sciences, Lunar and Deep Space Exploration Research Department, Beijing
[2]University of Chinese Academy of Sciences, Beijing
[3]Key Laboratory of Lunar and Deep Space Exploration, Chinese Academy of Sciences, Beijing
[4]Beihang University, Beijing

Abstract: A simulation system for lunar close-range exploration has been constructed for real-time lunar surface visualization and payload motion simulation. Based on the open-source Object-Oriented Graphics Rendering Engine (OGRE), combined with 3DMAX modeling software, Visual Studio integrated development tools, and terrain generation plugins, the system completes the construction of three-dimensional entity models for the lander and related payloads while providing realistic simulation of the lunar surface environment. The open-source CEGUI graphics library is employed to design a parameter panel for data display and human-computer interaction. Additionally, a three-dimensional scene debugging interface is built upon the Microsoft Foundation Classes (MFC) library to facilitate direct control of 3D scene updates and adjustment of entity model attitudes, enabling simulation of payload detection processes. Finally, simulation experiments are conducted using the panoramic camera as an example.

Keywords: Visual simulation; Lunar surface exploration; Model building; OGRE; Panoramic camera

1. Introduction

With the successful implementation of the Chang' e-1, Chang' e-2, and Chang' e-3 missions, China has accomplished the first phase of its lunar exploration program and is currently undertaking the third phase, which aims to achieve automatic unmanned lunar sampling and return. The primary mission is undertaken by the Chang' e-5 lunar probe, whose objectives include on-site investigation and analysis of the landing area, as well as analysis and research of lunar samples after their return to Earth. To achieve these goals, Chang' e-5 is equipped not only with engineering payloads such as a sampling robotic arm and monitoring cameras to provide contextual information for the returned samples, but also with scientific payloads including a panoramic camera, lunar mineralogical spectrometer, and landing camera for close-range lunar surface exploration.

During close-range exploration, the spatial environment, geometric relationships between payloads, and constraints and influencing factors of scientific detection all impact the results. Factors such as illumination conditions and terrain environment, as well as the three-body relationship between the Sun, Earth, and Moon, affect the detection outcomes. Since the scientific exploration process is unique and cannot be repeated, ground simulation experiments that model the lunar spatial environment beforehand play a crucial auxiliary role. By converting simulation data into three-dimensional graphical representations that visualize the process as it varies in time and space, researchers can grasp every detail of the experiment in real time, conduct multiple simulation trials, compare results, and make corresponding adjustments to the in-situ detection process. The repeatable nature of simulation technology enables optimization of detection plans and provides important basis for final mission decision-making.

To address questions of how to detect, what targets to select, and how to display detection results during lunar close-range exploration, we designed a simulation system capable of real-time 3D scene updates, detection effect display, and process playback. All payload scientific detection operations within the system can be directly controlled through the developed 3D scene debugging interface. The debugging interface enables rapid simulation of scientific detection processes, mission planning, and detection process playback. Users can independently write detection process script files according to their needs, enabling complete demonstration of the lunar in-situ detection workflow and results, and achieving high-demand, high-speed human-computer interaction.

2. System Design

2.1 System Functional Requirements Current international 3D visual simulation systems are mostly based on OpenGL, such as the flight training system and real-time flight simulation system developed by NASA' s Dryden Flight Research Center, and the 3D visual software MultiGen-Paradigm Creator/Vega. The University of Washington developed a mobile robot visual simulation system based on the Object-Oriented Graphics Rendering Engine, while Queensland University of Technology in Australia developed a robot visual simulation system based on OpenGL and DirectX.

The lunar close-range exploration simulation system is built upon OGRE (Object-Oriented Graphics Rendering Engine) combined with Visual Studio. While using graphics libraries like OpenGL and DirectX offers certain advantages, employing geometric primitives for complex 3D scenes requires extensive code writing. The object-oriented graphics rendering engine, rooted in object-oriented principles, not only abstracts and encapsulates all usage details of underlying interfaces (OpenGL, Direct3D) and other classes, but also provides convenient interfaces based on real-world objects. It supports rendering to texture, multi-level-of-detail (LOD) technology, scene management, and advanced particle system plugins, well satisfying the requirements for lunar close-range exploration from mission planning to scientific detection and result display. With high efficiency, multi-platform and 3D rendering API support, and high scalability for data formats and materials, these advantages enable the system to conveniently and quickly implement the following functions:

- **Time lapse rate adjustment:** Set system internal time resolution and dynamically change display rate according to user requirements.
- **Spatial position update:** Dynamically update the Sun-Earth-Moon spatial positions based on ephemeris data and complete 3D dynamic display.
- **Specific payload close-range detection:** Implement human-computer interaction through parameter panels and 3D scene debugging panels based on instruction script files and lander payload parameters to simulate the lunar close-range detection process.

- **Detection process playback:** Record operational processes in a custom format and save as script files for playback at any time.
- **Custom script file writing:** Implement user-required scientific detection simulation by calling script files.

2.2 System Overall Framework Following modular design principles, the lunar close-range exploration simulation system consists of three main modules: model building, simulation driving, and human-machine interface. Each module is independent yet interrelated, with relatively concentrated internal functions, facilitating functional expansion and maintenance. The overall framework is shown in [Figure 1: see original paper].

The model building module includes construction of 3D entity models and lunar surface terrain models. The simulation driving module primarily implements updates to 3D entity model attitudes and Sun-Earth-Moon spatial positions through the 3D scene debugging interface. The human-computer interaction module mainly refers to the parameter panel display interface and mouse input.

2.3 System Simulation Process The lunar close-range exploration simulation system employs a 3D graphics engine based on scene graph technology. When running the Object-Oriented Graphics Rendering Engine, the process includes: (1) Scene initialization, including creating root objects, parsing resource configuration files, creating initial scenes, and creating cameras and viewports; (2) Frame loop, completing each frame’ s rendering and message response. The system is designed with strong flexibility and scalability, allowing optional loading of plugins or library files based on functional requirements. The specific simulation flow is shown in [Figure 2: see original paper].

3. System Implementation

During initialization of the Object-Oriented Graphics Rendering Engine and resource files, relevant scene parsing libraries, CEGUI libraries for parameter interfaces, Microsoft Foundation Classes for 3D scene debugging, and ephemeris databases for Sun-Earth-Moon spatial position updates are loaded simultaneously. The system also continuously monitors user messages to obtain simulation control instructions and real-time motion data.

3.1 Model Building Model building primarily includes lunar surface terrain model construction and 3D entity model construction.

3.1.1 Lunar Surface Terrain Model Building The Object-Oriented Graphics Rendering Engine itself does not provide 3D model creation functionality. The model format used is .mesh, which belongs to a mesh model. Terrain rendering is one of the main scientific objectives of lunar exploration activities. To facilitate verification of system simulation results, we used orthoimage data with resolution and elevation data with spatial resolution obtained from ground validation tests to construct terrain models as substitutes for lunar surface terrain.

First, the terrain' s digital elevation model data and orthoimage data undergo pyramid processing to generate multi-level-of-detail (LOD) models. Considering the need for smooth rendering of large-scale terrain data, we adopted a terrain drawing algorithm using the .mesh file format. Based on terrain complexity and human visual characteristics, different regions are described and rendered with different levels of detail. This algorithm minimizes the number of triangles without reducing visual quality, thereby improving graphics rendering efficiency and enabling real-time interactive visualization of terrain. In simple terms, a terrain block is divided into numerous small regions: regions closer to the viewpoint or with more complex terrain are rendered with more triangles and higher precision, while distant or flatter regions use fewer triangles and lower precision. [Figure 3: see original paper] shows the established terrain model from ground validation experiments.

3.1.2 3D Entity Model Building Since the system is used for post-landing lunar scientific detection, Chang'e-5 payloads mainly include the lander, national flag, and sampling robotic arm. Payload models use real model parameters and proportions, with reference to panoramic camera gimbal motion videos and lunar mineralogical spectrometer dust cover opening videos, as well as robotic arm surface sampling and placement motion videos.

The Sun and Earth models are regular spheres implemented through simple texture mapping. 3DMAX, the most popular 3D animation and modeling software on personal computers, integrates rich third-party plugins and supports multiple formats. However, this format cannot be directly supported by the Object-Oriented Graphics Rendering Engine. Through a third-party plugin 3ds2mesh, the format is converted to the .mesh format recognizable by the engine.

3.2 Simulation Driving Simulation driving primarily implements entire 3D scene updates through the 3D scene debugging interface, including 3D entity model attitude control and Sun-Earth-Moon spatial position updates. Additionally, script-driven simulation is also supported.

3.2.1 MFC-Based 3D Scene Debugging Interface The Microsoft Foundation Classes (MFC) library encapsulates most WINDOWS system API functions and includes an application framework. We designed a 3D scene debugging interface based on MFC, which is of great significance for the entire system simulation. During lunar close-range exploration, this interface offers substantial advantages: it can quickly control entity model attitudes within the 3D scene, record each detection process, save the records, and playback the detection process by calling script files. Additionally, through the debugging interface, users can directly set system time, adjust time lapse rates, or change time points. The specific interface design is shown in [Figure 4: see original paper].

The entire interface is divided into two parts: simulation data recording and data playback. The simulation data recording section includes attitude control for the lander and various payloads, selection options, and time adjustment.

3.2.2 Sun-Earth-Moon Spatial Position Update The Microsoft Foundation Classes store raw Sun-Earth-Moon position data as time-indexed database files. According to the input JPL DE421 precision planetary ephemeris file path, the ephemeris database is called to calculate the Sun and Earth' s azimuth and elevation relative to the lunar observation point at a given time. As the system runs, Sun-Earth-Moon spatial position updates are mainly manifested in solar elevation angles and shadow effects.

3.2.3 Custom Script File Writing Script-driven simulation is achieved by writing custom script files through the 3D debugging interface to drive entire 3D scene updates. Drawing from Chang' e-5 lunar surface operational procedures, the script file content is shown in [Figure 5: see original paper]. The first column represents the date, the second column represents time, the third column represents the payload, and the fourth column represents payload attitude, with columns separated by spaces.

By calling the written script file, the simulation results shown in [Figure 6: see original paper] are obtained. Assuming the lander confirms safe landing at 02:05:45 on January 1, 2018, the lunar mineralogical spectrometer dust cover opens and begins detection at 02:00:45; the lander solar panels deploy to their initial position at 04:23:25 on January 2, 2018; the national flag deploys at 04:16:55; and finally, the panoramic camera begins autonomous detection.

3.3 Human-Computer Interaction The parameter panel display interface is based on CEGUI (Crazy Eddie' s GUI System), which uses object-oriented design and visual methods. It is a free, open-source graphics user interface library that supports multiple graphics libraries. The official LayoutEditor provides an interface editor that generates layout files in .layout format. By placing these files in the Object-Oriented Graphics Rendering Engine resource directory, the interface can be loaded directly. Users can freely select which parameter interface to display, with the F7 and F12 keys controlling parameter panel visibility. The parameter panel interface designed for the system is shown in [Figure 7: see original paper].

There are two modes for viewpoint control: (1) When binding to the lander, the lander remains at the screen center, so the entire space environment moves with it; (2) When binding to the panoramic camera, the camera always remains at the screen center, and other space environment changes occur with the camera' s movement. This local viewpoint design more clearly and directly displays simulation results and facilitates close-range detection simulation experiments for the panoramic camera. When selecting free viewpoint, users can arbitrarily choose viewpoints of interest. The entire lander can be zoomed, scaled, and rotated using the mouse, and the "W, A, S, D" keys control the lander' s distance and orientation.

4. System Application

Lunar close-range exploration simulation primarily concerns payload detection processes, detection effects, and detection mission simulation. Considering the influence of illumination effects and to better reflect visual simulation characteristics, we selected the Chang' e-5 panoramic camera for simulation experiments. During simulation, the system can output simulation time, solar elevation angle, solar azimuth, payload attitude, and other parameters. The output data can be used for scientific analysis and result verification.

4.1 Panoramic Camera Coverage Analysis The panoramic camera is designed to obtain high-resolution lunar surface images of the landing area and sampling zone for terrain and geomorphology research. Due to factors such as the lander' s own obstruction, related component occlusion, and the panoramic camera gimbal' s motion range, the stereo images acquired by the camera can only cover the southern half of the Chang' e-5 landing area. The panoramic images covering the area within a horizontal range of the landing point are called panoramic camera lunar surface panoramic imaging coverage areas.

Assuming the panoramic camera' s initial position is at (0°), with a total of 15 imaging positions requiring pitch angles of 0°, -12°, ···, -84° and yaw angles of -84°, -72°, ···, -12°, 0°, 12°, ···, 72°, 84° for each pitch corresponding half-circle. However, the image area obtained at this time is not the maximum lunar surface coverage area. shows the relationship between lander equivalent inclination angle and panoramic camera imaging strategy.

In the system environment, panoramic camera attitude can be adjusted according to actual conditions to obtain high-resolution lunar surface coverage images. By inputting panoramic camera attitude parameters, the system simulates the image acquisition process and sequentially outputs stereo image pairs. [Figure 8: see original paper] shows the panoramic camera imaging strategy under ideal horizontal and non-ideal conditions. [FIGURE:9(a)] shows partial image coverage within the maximum coverage area of panoramic camera panoramic imaging, while [FIGURE:9(b)] shows the red-green stereo image synthesized from the images.

4.2 Imaging Geometry Parameter Calculation To ensure high signal-to-noise ratio in detection data, illumination conditions during detection must be evaluated to determine whether payloads can be activated. Due to bidirectional reflectance characteristics varying under different illumination conditions, solar elevation angle, azimuth angle, and phase angle during imaging must be obtained for photometric correction processing.

Solar azimuth angle is generally measured clockwise from true north of the target to the direction of sunlight incidence. Solar elevation angle is the angle between sunlight and the lunar plane. The phase angle θ is the angle between incident sunlight and emergent light from the imaging target. [Figure 10: see original paper] shows a schematic diagram with the cross representing panoramic camera position, star representing imaging target, α as solar azimuth angle, β as solar elevation angle, and θ as phase angle.

Using JPL DE421 ephemeris data, we calculated the solar azimuth and elevation angles at the Chang' e-5 pre-selected landing point over one day. The variations during lunar daytime are shown in [FIGURE:11(a)] and [FIGURE:11(b)].

With the panoramic camera' s initial position facing south (target true north direction), we establish a coordinate system with the imaging target as origin, true east as x-axis, true north as y-axis, and vertical to lunar plane upward as z-axis. The incident sunlight direction vector is unit vector A, and emergent light direction vector is unit vector B. According to geometric relationships:

$$\theta = \arccos[\cos(\alpha)\cos(\beta)\cos(\mathrm{p})\cos(\mathrm{f}) + \cos(\beta)\sin(\mathrm{p})\cos(\mathrm{f}) + \sin(\beta)\sin(\mathrm{f})]$$

where α is solar azimuth angle, β is solar elevation angle, p is panoramic camera yaw angle, and f is panoramic camera pitch angle absolute value.

[Figure 12: see original paper] shows the θ value variations corresponding to different panoramic camera attitudes. Given the panoramic camera activation condition (solar elevation angle unchanged), it shows θ values under different attitudes at the same moment. The phase angle θ changes with panoramic camera attitude: when pitch angle is constant, phase angle increases with yaw angle; within one yaw cycle, phase angle decreases with increasing pitch angle. When panoramic camera attitude is fixed, θ changes with solar elevation angle.

Based on the solar azimuth and elevation data above, we calculated θ values under different solar elevation angles during lunar daytime when the camera attitude is fixed. The phase angle is positively correlated with solar elevation angle within a certain range, and negatively correlated beyond that range. When pitch angle is small and imaging range is wide, phase angle and solar elevation angle show the same trend. When pitch angle is large and imaging range becomes smaller, their trends become opposite, with larger phase angles.

4.3 Detection Effect Simulation Using this simulation system, we simulated panoramic camera detection effects under different solar elevation angles. [FIGURE:14(a)] and [FIGURE:14(b)] show images taken at different θ angles, demonstrating that image quality varies with θ angle, indicating that solar illumination affects panoramic camera performance.

5. Conclusion

As lunar exploration projects advance, research on interactive 3D visualization simulation systems has clear practical significance. The constructed lunar close-range exploration simulation system can perform real-time visual demonstration of the Chang’ e-5 lander and is not limited to Chang’ e-5 alone—it is equally applicable to other payloads. The system’ s 3D scene debugging interface can directly control 3D scene updates and payload attitudes, enabling simulation of payload detection processes and missions. The greatest advantage of the system lies in its modular design with strong flexibility and scalability, where each module is independent yet complementary. By modifying module contents, it can also be adapted for close-range exploration of other planetary surfaces.

We verified the system’s correctness through panoramic camera panoramic imaging coverage simulation and analysis, comparing results with actual panoramic camera experiment images. Of course, the system still has room for improvement. For instance, it could be integrated with 3D stereo hardware to develop active stereo display using binocular stereo display technology, which enables human eyes to obtain stereo vision on 2D display planes.

References

[1] Ye Peijian, Sun Zezhou, Rao Wei. Research and development summary of Chang’ e-1 lunar exploration satellite. Spacecraft Engineering, 9-15.

[2] Pan Chen, Pang Dan, Wu Xuan. Chinese lunar exploration gives people another surprise (Part 1)—Perspective on the reentry return flight test mission of the third phase of lunar exploration program. Aerospace China.

[3] Ouyang Ziyuan. Academician Ouyang Ziyuan outlines the follow-up blueprint for the Chang’ e program. Science Times.

[4] Zeng Xin. Design and implementation of visual simulation technology. Heilongjiang Science and Technology Information.

[5] Le J H. The role of aircraft simulation in improving flight safety through control training. California National Aeronautics and Space Administration Dryden Flight Research Center.

[6] Shy K S, Hagemann J J, Curllett B P. A software framework for aircraft simulation.

[7] Faust J, Simon C, Smart W D. A video game-based mobile robot simulation environment. 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, 3749-3754.

[8] Dunn T L, Wardhani A. A 3D robot simulation for education. Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia, 277-278.

[9] Huang Weiquan, Wang Xiaoming, Chen Huaiyou. Design and implementation of a missile visual simulation system. Computer Simulation, 252-255+259.

[10] Wang Yaming, Su Yan, Feng Jianqing, et al. A review of lunar polar exploration. Astronomical Research & Technology, 300-309.

[11] Zong Qun, Xu Rui, Li Qingxin. A real-time visual simulation system for UAV full trajectory. Control Engineering of China, 32-33.

[12] Lei Junhuan, Zeng Fanxi, Wu Mingxing. Research on LOD algorithm based on quad-tree structure. Manufacturing Automation, 211-214+228.

[13] Autodesk 3ds Max 2010 standard training materials. Beijing: Posts and Telecom Press.

[14] Xu Xuesen, Liu Jianjun, Liu Bin, et al. Progress on research of lunar photometric model. Remote Sensing Technology and Application, 634-644+652.

[15] Gao Xingye, Liu Jianjun, Ren Xin, et al. Application of binocular stereo display technology in three-dimensional visualization of lunar surface morphology. Astronomical Research & Technology, 358-365.

[16] Li Weifeng. Study on 3D terrain visualization based on red-green stereo images—Stereo display of terrain digital image pairs.

[17] Zheng Chen. Research and implementation of key technologies in 3D simulation for electronic countermeasures. Chengdu: University of Electronic Science and Technology.

[18] Wang Yaming. Research and implementation of visual simulation technology based on graphics rendering engine. Nanjing: Nanjing University of Aeronautics and Astronautics.

Note: Figure translations are in progress. See original paper for figures.