HARMONICS: A Visualization Tool for Hayabusa and Hayabusa 2 Missions*

Wataru UENO,† Hirohide DEMURA, and Naru HIRATA

Department of Computer Science and Engineering, The University of Aizu, Fukushima 965–8580, Japan

We developed a tool for visualizing the spatial geometry of objects and field-of-view (FOV) of scientific instruments for mission plans and data analysis of Hayabusa and Hayabusa 2, and named “HARMONICS (Hayabusa Remote MONitoring and Commanding System).” We also implemented a graphical user interface to simulate a changing FOV. Displaying arbitrary viewpoints over a time sequence helps determine the geometry observed and supports later data analysis. HARMONICS loads ancillary data with the SPICE kernel format: position and attitude of the spacecraft, properties of scientific instruments and target’s shape model, etc. Here, we report on the system details and enhanced functions of HARMONICS compared to the original version in 2005.

Key Words: Hayabusa 2, SPICE, Flight Simulator, Navigation, Space Exploration

1. Introduction

In deep-space explorations beyond the moon, 3D visualization with arbitrary viewpoints is useful. When scientists request mission plans with a mission scenario, procedures should be supported by visualizing a field-of-view (FOV).1 Scientists consider observation periods that cover targets, positions/attitudes of the spacecraft, and required conditions such as the illumination angle and spatial resolution at a certain time. This tool supports not only beginners making plans, but also scientists checking observation data closely, such as verifying a simulation by comparing it with observations. In other cases, this tool can also be used to simulate spatial configurations through a trial-and-error process.

Here, we describe a visualization tool named “HARMONICS (Hayabusa Remote MONitoring and Commanding System)” for planning and scientific analysis of Hayabusa/Hayabusa 2 missions2,3 to asteroids. We developed HARMONICS to plan missions during the rendezvous phase in the Hayabusa and Hayabusa 2 missions. Because the maneuvers of Hayabusa-type spacecraft do not follow a periodic orbit, such a tool for visualizing arbitrary viewpoints is required to plan the observations conducted by scientific instruments with the spacecraft’s changing position and attitude. HARMONICS helps science team members efficiently consider mission plans for scientific instruments based on maneuvers of the spacecraft. HARMONICS was successfully used as a scientific analysis and planning tool for the Hayabusa mission in 2005.4–7 Figure 1 shows the original version. This tool was also applied to verifying ancillary data (positions relative to the targeted asteroid) and obtaining shape models by comparing images observed with images rendered based on the shape model.4–7 HARMONICS was also used by scientists to analyze image data after the rendezvous operations. HARMONICS will support the Hayabusa 2 mission during its rendezvous phase in 2018.

Although similar tools have begun to appear in deep-space explorations, they are not intended for practical use in our purpose.

2. Two Use Cases of HARMONICS

2.1. Planning missions

Planning missions is the first use case of HARMONICS, as shown in Fig. 2. This figure is drawn according to the Project Management Body of Knowledge (PMBOK) style.8 Users can determine the FOV simulation at a certain time and search for the best geometry and observation timing based on the spacecraft’s changing position and attitude. In the first step, the user loads a sequence of positions and attitudes for the spacecraft as an initial mission plan into HARMONICS. Next, the user displays time-sequential FOVs for the scientific instruments and geometry of the spacecraft and target in order to determine the observation timing. The user can select onboard scientific instruments and compare FOVs. Third, the user can slightly modify the position and attitude (e.g., pointing direction) of the spacecraft in HARMONICS if there is a need to change the pre-defined mission plan. This tool simulates the FOVs of the onboard scientific instruments and geometry of the spacecraft and target based on the modified position and attitude of the spacecraft in real-time, and the user can record the results. The user can survey the effects of the modified positions and attitude of the spacecraft. Finally, the user exports the modified position and attitude of the spacecraft as a new mission plan. The user repeats the above steps to develop an appropriate mission plan for the spacecraft.

2.2. Determining the spatial configuration during observation

The second use case of HARMONICS is determining the spatial configuration during observation, as shown in Fig. 3.
functions were implemented in HARMONICS in 2005. The third, fourth, and fifth functions are new functions of the current HARMONICS.

3.1. Displaying of the simulation results of a scientific instrument's FOV

HARMONICS loads the position and attitude of the spacecraft from datasets of the Hayabusa and Hayabusa 2 missions. The tool simulates the FOV of a scientific instrument on the spacecraft from the observation parameters input, position and attitude of the spacecraft, and information about onboard scientific instruments in a time series animation. The user plans activities for the scientific instruments based on the simulation results for the FOV. The user also visualizes spatial configurations during observations with 3D graphics for observation data analysis.

3.2. Visualization of the geometry between the spacecraft and target from a third-person viewpoint

The user plans missions for the spacecraft while verifying the locations of the Earth and Sun. In the observation data analysis phase, the user simulates the geometry between the spacecraft and target with HARMONICS to determine the illumination conditions.

3.3. Selection of the onboard scientific instruments

In 2005, HARMONICS supported only an optical navigation camera on the spacecraft. The new HARMONICS supports almost all scientific instruments: a spectrometer (NIRS3), laser altimeter (LIDAR), thermal camera (TIR), etc., and is capable of simulating the FOVs and boresights of these scientific instruments.

3.4. Simulation based on user input

Users could not interactively and precisely change the position and attitude of the spacecraft model with HARMONICS in 2005. Now, users can load the position and attitude of the spacecraft from files containing ancillary data into HARMONICS. A user is required to overwrite the datasets directly to make fine adjustments to the position and attitude of the spacecraft model, doing so by entering values for the spacecraft model's position and attitude into the interface of HARMONICS. This function helps a user efficiently plan the best activities for onboard scientific instruments.

3.5. Export of the position and attitude of the spacecraft as a new mission plan

Users usually simulate spacecraft missions repeatedly to
plan appropriate activities for the onboard scientific instruments. A user can modify the position and attitude of the spacecraft during an initial mission plan, export the modified position and attitude, and reload the modified position and attitude as a new mission plan.

3.6. Display a comparison of the images observed from the spacecraft with the simulation results

HARMONICS loads images observed from the spacecraft using major image formats such as JPEG, PNG, and FITS. This tool receives the observation time from the loaded images and then simulates the FOV of the onboard scientific instruments and geometry between the spacecraft and target during this observation time.

4. System Design

Figure 4 shows the module structure of HARMONICS. HARMONICS consists of three modules. It adopts the GUI library Qt,9) which is presented in Section 4.1. The user interface of HARMONICS follows that of the HARMONICS used in 2005. HARMONICS receives the position and attitude of the spacecraft and information about onboard scientific instruments from datasets of the Hayabusa or Hayabusa 2 missions using SPICE,10) as discussed in Section 4.2. Then, the tool renders the FOV of onboard scientific instruments and geometry between the spacecraft model and that of the asteroid targeted OpenGl, the 3D graphics library,11) as discussed in Section 4.3. The tool displays the scene on the GUI. When the user modifies the position and attitude of the spacecraft in HARMONICS, the tool simulates the FOV of the onboard scientific instruments and geometry between the spacecraft and target based on the modified parameters of the spacecraft results in real-time. We describe the three modules of HARMONICS in the following subsections.

4.1. GUI module powered by Qt

The HARMONICS interface was developed using the cross-platform GUI library, Qt. HARMONICS of 2005 was developed using the GTK, GUI library,12) GTK and its descendant, GTK+, show incompatibility among their versions.13) We surveyed preferable GUI libraries from the viewpoint of lifetime, and selected the candidates; Qt,9) Swing14) and WxWidgets15) in Table 1. We adopted Qt for HARMONICS because of its compatibility with other library versions and drawing functions in graphs/charts. HARMONICS works on major operating systems: Windows, MacOS, and UNIX/Linux. Although Swing is a cross-platform library, it has incompatibility with some versions. The GUI of WxWidgets has poor usability. It behaves differently on each platform.

4.2. Geometry computation module powered by SPICE

NASA/JPL provides an information system called SPICE to assist users with planning and interpreting spacecraft missions. SPICE is used for the majority of planetary explorations around the world. The SPICE toolkit17) was developed by NASA/JPL to handle SPICE kernels,18) and provides both application programming interfaces (APIs) and standalone utilities. SPICE kernels include information about the positions of spacecraft and planets, the attitude of the spacecraft, onboard scientific instruments, reference frames, and the rotation period of planets. Developers can integrate the SPICE toolkit into their applications to calculate observation parameters, the position and attitude of the spacecraft, the illumination conditions of the target’s surfaces, etc. HARMONICS loads SPICE kernels and then calculates the observation geometry parameters using the SPICE toolkit. HARMONICS can visualize the FOV of scientific instruments on Hayabusa or Hayabusa 2 and the geometry of the spacecraft and target based on the parameters loaded. HARMONICS exports the modified position and attitude of the spacecraft using the SPICE kernel format.19) The exported kernels can be utilized as a new mission plan based on the utilities of the SPICE toolkit.

4.3. 3D model renderer powered by OpenGl

OpenGL is a powerful graphic library for producing high-quality images of 3D objects.20) The library performs 2D/3D rendering in real-time. OpenGL has powerful functions, rendering, texture mapping, etc. This library is widely used for supporting 2D/3D graphics. HARMONICS displays the spacecraft and asteroid shape with OpenGl.

5. Results

HARMONICS follows all of the required functions. HARMONICS can load the SPICE kernels and then simulate the FOV of scientific instruments on either Hayabusa or Hayabusa 2 and the geometry between the spacecraft and...
target based on the SPICE kernels loaded. The user can plan a mission based on the images rendered by HARMONICS (e.g., see Fig. 5).

HARMONICS is a cross-platform tool. This tool works on major operating systems such as MacOS, Windows and UNIX/Linux (see Fig. 6). Users do not have to prepare computing environments for HARMONICS.

A user can interactively make fine adjustments of the position and attitude of the spacecraft using the tool and then export the modified position and attitude of the spacecraft in the SPICE kernel format as a new mission plan.

HARMONICS is used by scientists to determine spatial configurations during observations. The tool reads the observed position and attitude of the spacecraft, information about the onboard scientific instruments, etc., and then simulates the FOV of the onboard scientific instruments or geometry between the spacecraft and target during observation using 3D graphics. The user can also determine the illumination conditions during observations. HARMONICS can display a comparison of the images observed from the spacecraft with the simulation results. HARMONICS helps users find imaged areas of the spacecraft and analyze observation data from the spacecraft.

In the first step, the tool loads ancillary data in the SPICE kernel format: the position and attitude of the spacecraft, information about onboard scientific instruments, etc. In the second step, the tool computes the observation parameters, position and attitude of the spacecraft, and illumination conditions. In the third step, HARMONICS displays the observation parameters shown in Fig. 7(c)-iv, the FOV of the onboard scientific instruments, and geometry with the spacecraft and asteroid shape models, as shown in Fig. 7(a). In addition, the user can select onboard scientific instruments (see Fig. 7(c)-ii). Users can interactively make fine adjustments to the position and attitude of the spacecraft model (see Fig. 7(c)-iii). If the user modifies the position and attitude of the spacecraft in HARMONICS, the tool simulates the FOV of the onboard scientific instruments and geometry between the spacecraft and target based on the modified position and attitude of the spacecraft in real time. The tool records the modified position and attitude of the spacecraft and then exports the modified parameters of the spacecraft in the SPICE kernel format. HARMONICS reloads SPICE kernels that contain the modified position and attitude. The user repeats the above steps to develop an appropriate mission plan for Hayabusa 2.

6. Discussion and Summary

This paper summarizes and explains the enhanced functions of HARMONICS compared to the original version in 2005. Science team members for Hayabusa 2 can examine and make mission plans and export modified mission plans with some trial-and-error using simulated images from HARMONICS. Further, scientists can reproduce the observation geometry and operation sequences, which will help with analyzing the datasets of Hayabusa and Hayabusa 2 using HARMONICS.
The new function of exporting the modified mission plan as SPICE kernels is implemented by external commands, which are ready-built utilities in the SPICE toolkit. HARMONICS shows high potential to be adapted to other space programs that use SPICE kernels.

Simulated images in HARMONICS rendered using OpenGL do not correctly represent the shadows casted (Fig. 8). This problem does not greatly affect mission planning and scientific analysis. In mission planning, scientists can roughly infer shadow regions on a target with landforms and the sun direction that HARMONICS simulates. In scientific analysis phase, scientists want to simulate images of Hayabusa and Hayabusa 2 observation, position/attitude of the spacecraft and the direction of the sun. However, HARMONICS needs to output more simulation results to enable accurate mission planning and scientific analysis. We need to use another shadow rendering technique called “ray tracing” for users to more closely simulate the actual FOV of onboard scientific instruments and geometry between the spacecraft and targets. Although ray tracing is popular for generating a high-quality image in a 3D graphics field, it requires an enormous computational cost. We will reduce the computational cost of ray tracing and then implement the method in HARMONICS.

Acknowledgments

We would like to express special thanks to Ms. E. Nemoto as the original HARMONICS developer and Dr. R. Nakamura as an advisor in 2005. We gratefully acknowledge the many users of the Joint Science Team for the Hayabusa mission and others. The authors thank the anonymous reviewers for their helpful suggestions. We would also like to thank Editage (www.editage.jp) for English language editing. This work was supported by JSPS KAKENHI Grant Number JP17K05639.

References

1) Acton, H., Bachman, N., Semenov, B., and Wright, E.: SPICE Supports Planetary Science Observation Geometry, American Astronomical Society, DPS Meeting #47, id # 312.05, 2015.
2) Yoshikawa, M., Kawaguchi, J., Fujisawa, A., and Tsujiyama, A.: PART 2: SPACE MISSIONS, “Hayabusa Sample Return Mission”, Michel, P., DeMee, F. E., and Bottke, W. F., eds., Asteroids IV, University of Arizona Press, Tucson, 2015, pp. 397–418.
3) Yoshikawa, M., Watanabe, S., Tsuda, Y., Kuminaka, H., and Hayabusa2 Project Team: Hayabusa2–The Next Asteroid Sample Return Mission of Japan, Trans. JSASS Aerospace Technology Japan, 12, istu29 (2014), pp. Tk. 29–Tk.33.
4) Demura, H., Kobayashi, S., Nemoto, E., Matsumoto, N., Furuya, M., Yukishita, A., Muranaka, N., Morita, H., Shirakawa, K., Maruya, M., Ohyama, H., Uo, M., Kubota, T., Hashimoto, T., Kawaguchi, J., Fujiwara, A., Saito, J., Sasaki, S., Miyamoto, H., and Hirata, N.: Pole and Global Shape of 25143 Itokawa, Science, 312 (2006), pp. 1347–1349.
5) Saito, J., Miyamoto, H., Nakamura, R., Ishiguro, M., Michikami, T., Nakamura, M., Demura, H., Sasaki, S., Hirata, N., Honda, C., Yamamoto, A., Yokota, Y., Fuse, T., Yoshida, F., Tholen, D., Gaskell, R., Hashimoto, T., Kubota, T., Higuchi, Y., Nakamura, T., Smith, P., Hiraoka, K., Honda, T., Kobayashi, S., Furuya, M., Matsumoto, N., Nemoto, E., Yukishita, A., Kitazawa, K., Dermawan, B., Sogame, A., Terazono, J., Shinohara, C., and Akiyama, H.: Detailed Images of Asteroid 25143 Itokawa from Hayabusa, Science, 312 (2006), pp. 1341–1344.
6) Fujiwara, A., Kawaguchi, J., Yoemans, D., Abe, M., Mukai, T., Okada, T., Saito, J., Yano, H., Yoshioaka, M., Scheeres, D., Barnouin-Jha, O., Cheng, A., Hirohide, H., Gaskell, R., Hirata, N., Ikeda, H., Kominato, T., Miyamoto, H., Nakamura, A., Nakamura, R., Sasaki, S., and Usugi, K.: The Rubble-Pile Asteroid Itokawa as Observed by Hayabusa, Science, 312 (2006), pp. 1330–1334.
7) Nemoto, E.: In-flight Calibration of AMICA: a Multiband Imaging Camera on Hayabusa, Master’s Thesis, University of Aizu, 2006 (in English).
8) Project Management Institute, http://www.pmi.org/ (accessed 9 Jun 2016).
9) Qt, https://www.qt.io/ (accessed 23 Dec 2015).
10) About SPICE, https://naif.jpl.nasa.gov/naif/aboutspice.html (accessed 23 Dec 2015).
11) OpenGL, https://www.opengl.org (accessed 23 Dec 2015).
12) The GTK+ Project, http://www.gtk.org (accessed 23 Dec 2015).
13) Migrating from GTK+2.x to GTK+3.x., https://developer.gnome.org/gtk3/stable/gtk-migrating-2-to-3.html (accessed 27 Nov 2016).
14) Java Swing Tutorial, http://www.javatpoint.com/java-swings (accessed 23 Dec 2015).
15) wxWidgets, https://www.wxwidgets.org/ (accessed 23 Dec 2015).
16) Transition from Qt4.x to Qt5.x, https://wiki.qt.io/Transition.from.Qt.4.x.to.Qt5 (accessed 27 Nov 2016).
17) The SPICE Toolkit, http://naif.jpl.nasa.gov/naif/toolkit.html (accessed 23 Dec 2015).
18) SPICE Data (SPICE Kernels), http://naif.jpl.nasa.gov/naif/data.html (accessed 23 Dec 2015).
19) An overview of SPICE, http://naif.jpl.nasa.gov/pub/naif/toolkit/docs/Tutorials/pdf/individual/docs/03.spice.overview.pdf (accessed 23 Dec 2015).
20) OpenGL Overview, https://www.opengl.org/about/1 (accessed 13 Nov 2016).

J. Kawaguchi
Associate Editor