P. Panicucci, J. Lebreton, R. Brochard, E. Zenou, M. Delpech
Shadow-Robust Silhouette Reconstruction for Small-Body Applications
Journal of Spacecraft and Rockets, Vol. 60, N. 3, 2023, p. 812-828
doi:10.2514/1.A35444

Shadow-Robust Silhouette Reconstruction for Small-Body Applications

Paolo Panicucci* †
ISAE-SUPAERO, 10 Avenue Edouard Belin, 31400, Toulouse, France
CNES, 18 Avenue Edouard Belin, 31400, Toulouse, France
Airbus Defence & Space, 31 Rue des Cosmonautes, 31400, Toulouse, France

Jérémy Lebreton‡ and Roland Brochard§
Airbus Defence & Space, 31 Rue des Cosmonautes, 31400, Toulouse, France

Emmanuel Zenou¶
ISAE-SUPAERO, 10 Avenue Edouard Belin, 31400, Toulouse, France

Michel Delpech‖
CNES, 18 Avenue Edouard Belin, 31400, Toulouse, France

Small bodies exploration highlighted the need to develop new algorithms for deep space probes navigation. The estimation of the small body shape is a crucial step for relative navigation, mission planning, and gravity investigation. This paper develops a shape from silhouette algorithm that takes as input a series of images to construct a polyhedral shape for small bodies of the Solar System. First, the silhouette associated with the visible part of the body is extracted and modified to account for shadowing effects. Second, the shape is computed by intersecting the viewing cones, i.e., the cone whose apex is the camera center and whose vertices are the silhouette points. The algorithm output is a polyhedral shape whose reconstruction is robust to shadowing. The algorithm is validated by reconstructing the shape of Itokawa and Bennu from synthetic images simulated with Airbus Defence & Space's rendering engine SurRender™. Moreover, the algorithm is tested on real images from Hayabusa's approach. Results prove the capability to provide an efficient shape reconstruction for small body applications.

I. Nomenclature

$\mathbb{S}$	=	An object
$\mathbb{VH}(\mathbb{S})$	=	The visual hull of an object

*PhD Student, Complex System Engineering Department
†Currently: Postdoctoral Fellow, Dipartimento di Scienze e Tecnologie Aerospaziali, Politecnico di Milano
‡R&D Engineer, Sensor Processing Chain Department
§R&D Engineer, Sensor Processing Chain Department
¶Associate Professor, Complex System Engineering Department
‖GNC Senior Expert, Future Mission Engineering Department

$\mathbb{s}(\mathbb{S})$	=	The surface on an object
$\mathbb{sa}(\mathbb{S})$	=	The silhouette active surface of an object
$\mathbb{si}(\mathbb{S})$	=	The silhouette active surface of an object
$\mathbb{vh}(\mathbb{S})$	=	The surface on the visual hull of an object
$\mathbb{CH}(\mathbb{S})$	=	The convex hull of an object
$\mathbb{R}_c$	=	The Euclidean subspace which allow to recover the visual hull
$\mathbb{VH}(\mathbb{S}, \mathbb{R})$	=	The visual hull of an object as recovered from a subspace $\mathbb{R}$
$\mathbb{vh}(\mathbb{S}, \mathbb{R})$	=	The surface of the visual hull of an object as recovered from a subspace $\mathbb{R}$
$\boldsymbol{P}_i$	=	The ith vertex of the silhouette
ψ_i	=	The heuristic associated with $\boldsymbol{P}_i$
$[F]$	=	The fundamental matrix
$[K]$	=	The camera intrinsic matrix
$\boldsymbol{r}_{\text{cam}}$	=	The camera center position
C	=	The image center
$\boldsymbol{l}_{\boldsymbol{P}}$	=	The line of sight of point $\boldsymbol{P}$
f	=	The focal length
FoV	=	The field of view
s_x, s_y	=	The pixel extension in x and y
ϵ	=	The normalized Hausdorff distance
δV	=	The relative volumetric error

II. Introduction

Shape reconstruction is a crucial step for relative navigation [1, 2], mission planning [3, 4], and gravity investigation [5–7]. In current small-body missions, optical data is downlinked to Earth and analyzed on ground to construct a high-fidelity shape model through stereophotoclinometry (SPC). This technique, which relies on shape from motion, has been used since the beginning of small body exploration to construct high-fidelity shape models and albedo maps [8–10]. Unfortunately, it strongly relies on human intervention to control the solution convergence and validate the output shape. Moreover, the required computational time makes it unsuitable for onboard applications.

For navigation purposes, a lower-resolution model could give the same navigation performances by limiting the computational burden associated with the reconstruction of a high-fidelity model, which is composed of a high number of variables, i.e. vertices and the associated connectivity matrix. Thus, it is of crucial interest to develop shape reconstruction algorithms that estimate a low-resolution shape.

The algorithm presented in this work aims at reconstructing an approximation of a small body shape which would enable the use of LIDAR [2], model-based tracking [1, 11], and ΔV ranging [12].

The main contribution of the silhouette-based reconstruction algorithm relies on the computation of the shadow-robust silhouettes, which are modified versions of the silhouettes extracted from raw images. The silhouette is modified by considering the Sun illumination angle and by projecting the shadows in the image. 2D raytracing is used to identify how shadows are projected in the image. These shadow projections help to remove limbs not directly illuminated by the Sun. This ensures that only accurate limb information is given to the silhouette-based reconstruction algorithm. Thanks to this procedure, the algorithm is robust to adverse illumination conditions. Silhouettes are then backprojected in the 3D space and intersected to generate an approximation of the shape. Numerical results are presented and discussed to understand the algorithm performances and limitations.

III. Notation

In this paper the following notation is used:

- 3D vectors are in lower case bold text, such as $\boldsymbol{r}$, and 2D vectors are in upper case bold, such as $\boldsymbol{R}$.
- Vector initialization is performed with parenthesis, such as $\boldsymbol{b} = \left(\boldsymbol{a}^T \ \boldsymbol{a}^T\right)^T$
- $\mathcal{A} = \{a, \boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3\}$ is a 3D reference frame centered in a with axes $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, and $\boldsymbol{a}_3$. All the reference frames are right-handed.
- $\mathbb{A} = \{A, \boldsymbol{A}_1, \boldsymbol{A}_2\}$ is a 2D reference frame. This reference frame is centered in A with axes $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ which are orthonormal.
- The vector $\boldsymbol{r}$ expressed in the $\mathcal{S}$ reference frame is denoted ${}^{\mathcal{S}}\boldsymbol{r}$
- The vector $\boldsymbol{R}$ expressed in the $\mathbb{S}$ reference frame is denoted ${}^{\mathbb{S}}\boldsymbol{R}$

IV. Shape Reconstruction in Robotics

Several techniques exist to reconstruct an object's shape. The simplest approach is to triangulate 3D points to obtain a point-wise representation of the shape [13]. These 3D points can be then linked to form a mesh to have a model exploitable for navigation. The main limitation relies on the fact that the 3D point projections association in different images is strongly connected with feature description, extraction, matching, and tracking. Features are extracted with dedicated algorithms according to pixel intensities that are not linked with geometrical entities, such as small body limbs. Some descriptors, such as the Harris descriptor [14], exploit image data to identify intensity variations, such as limbs for dark background, from the image but it is not mandatory that their output represent novel geometrical information with respect to feature-based localization methods, as the ones developed in Panicucci [15] or Dor et al. [16].

Alternatively, shadow carving [17] relies on the computation of the shadows as observed in the image and, by knowing the position of the light source, the 3D shape is carved to reproduce consistent shadows. For space applications, this

technique has several drawbacks. First, shadow detection is a complex task as shadows and the dark background are hard to distinguish. Second, a good first guess of the shape must be already available to start the carving process and the shadow simulation. Third, shadow simulation in the images relies on ray tracing which is a computationally expensive procedure to be performed.

Otherwise, shape from silhouettes is a technique that enables reconstructing the shape by combining the information present in all the available silhouettes. Algorithms based on silhouette usually use a voxel representation of the shape to be estimated. Voxels are three-dimensional pixels. A voxel approach has been previously studied for small body exploration for both on-ground [18] and autonomous [19] applications. This technique foresees projecting a predefined volume, e.g., a cube, in all camera views and carving of the voxels not belonging to the silhouettes. An alternative approach relies on the intersection of the viewing cones [20–22]. The viewing cone is defined by the rays starting at the camera center and passing through the points of one silhouette. By projecting this cone in the other silhouettes, segments belonging to the object are gathered and the shape can be reconstructed. Baker and McMahon [23] presents a shape from silhouette algorithm based on viewing cones intersection as in Franco and Boyer' work [21]. In Baker and McMahon's work, no adverse illumination is considered as images are carefully selected to obtain a precise shape approximation. This is a hypothesis that limits the algorithm's applicability in an autonomous framework as shadow identification is a hard task due to the dark background. In these shape from silhouette algorithms, the intersection of the viewing cone is performed by backprojecting a limited number of 2D points - generally the vertices - to compute an approximated solution for the cones' intersection. This approximated solution can strongly deviate from the correct visual hull computation (see Section V for a formal definition of the visual hull) when the small body rotation axis is tilted with respect to the Sun-small-body direction or the approach direction.

This work aims not only to solve the problem of visual hull approximation by intersecting the viewing cones with fewer approximations but also to integrate the proposed algorithm with shadow-robust silhouettes computation, which has been investigated in previous studies [15, 24].

V. The Visual Hull

Before entering the details of the algorithm, it is important to define what can be reconstructed from a series of silhouettes. Silhouettes are formed by small body limbs and, as a consequence, no information is carried about points, or features, that lie inside the silhouettes. Following this reasoning, if a point has not been seen in one of the small body limbs, no information can be deduced about its position. Thus it cannot be reconstructed.

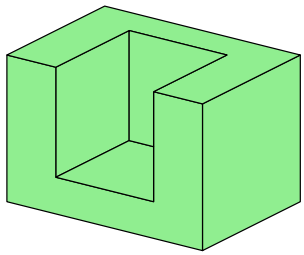
More rigorously, the volume that is possible to deduce from silhouettes is called the visual hull. The visual hull $\mathbb{VH}(\mathbb{S})$ of an object $\mathbb{S}$ is a concept formally introduced by Laurentini [25]. By using Laurentini's words, "[b]roadly speaking, the visual hull of an object $\mathbb{S}$ is the envelope of all the possible circumscribed cones of $\mathbb{S}$. An equivalent intuition is that the visual hull is the maximal object that gives the same silhouette of $\mathbb{S}$ from any possible viewpoint" [25]. It is

important to note that this definition implies an infinite number of cones, thus an infinite number of views.

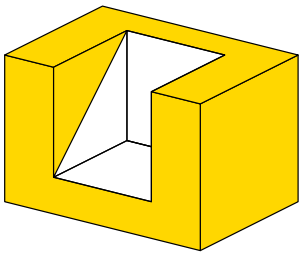
From the definition of the visual hull, it is clear that the object $\mathbb{S}$ is contained in the visual hull $\mathbb{VH}(\mathbb{S})$, i.e. $\mathbb{S} \leq \mathbb{VH}(\mathbb{S})$, as the cones tangency implies that some concavities cannot be captured even with an infinite number of views. From this consideration it is possible to divide the surface $\mathbb{s}(\mathbb{S})$ of the the object $\mathbb{S}$ in two subsets: the silhouette-active part $\mathbb{sa}(\mathbb{S})$ and the silhouette-inactive one $\mathbb{si}(\mathbb{S})$.

The silhouette-active surface $\mathbb{sa}(\mathbb{S})$ is the part of $\mathbb{s}(\mathbb{S})$ which is tangent to at least one cone. Therefore $\mathbb{sa}(\mathbb{S})$ also belongs to $\mathbb{vh}(\mathbb{S})$, the boundary of $\mathbb{VH}(\mathbb{S})$. The silhouette-inactive surface $\mathbb{si}(\mathbb{S}) = \mathbb{s}(\mathbb{S}) \setminus \mathbb{sa}(\mathbb{S})$ is, by definition, the not observable part of $\mathbb{s}(\mathbb{S})$ from silhouettes. This means that $\mathbb{si}(\mathbb{S})$ can assume any shape in $\mathbb{VH}(\mathbb{S}) \setminus \mathbb{S}$ without affecting the computation of $\mathbb{VH}(\mathbb{S})$. This implies that the object can be reconstructed from silhouette-based reconstruction if and only if $\mathbb{S} = \mathbb{VH}(\mathbb{S})$. Furthermore, another object $\mathbb{S}'$ can be discriminated from $\mathbb{S}$ if and only if $\mathbb{VH}(\mathbb{S}') \neq \mathbb{VH}(\mathbb{S})$. In Figures 1a - 1e examples of the previously-defined surfaces and volumes are shown.

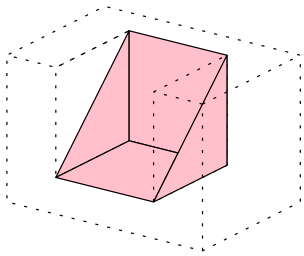
If an infinite number of cameras are placed outside the visual hull - as small body's approach - it can be stated that



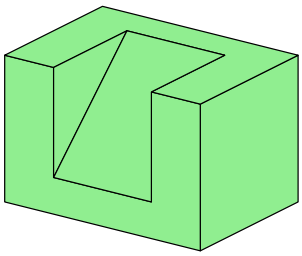
(a) The object $\mathbb{S}$ and its surface $\mathbb{s}(\mathbb{S})$



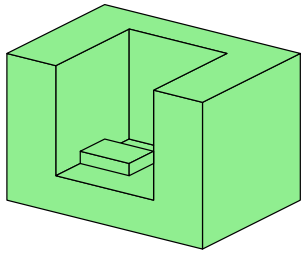
(b) The silhouette-active surface $\mathbb{sa}(\mathbb{S})$



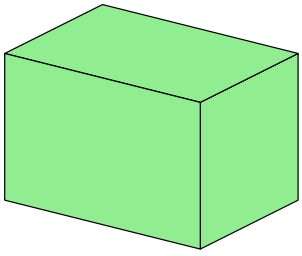
(c) The silhouette-inactive surface $\mathbb{si}(\mathbb{S})$



(d) The visual hull $\mathbb{VH}(\mathbb{S})$ and its surface $\mathbb{vh}(\mathbb{S})$

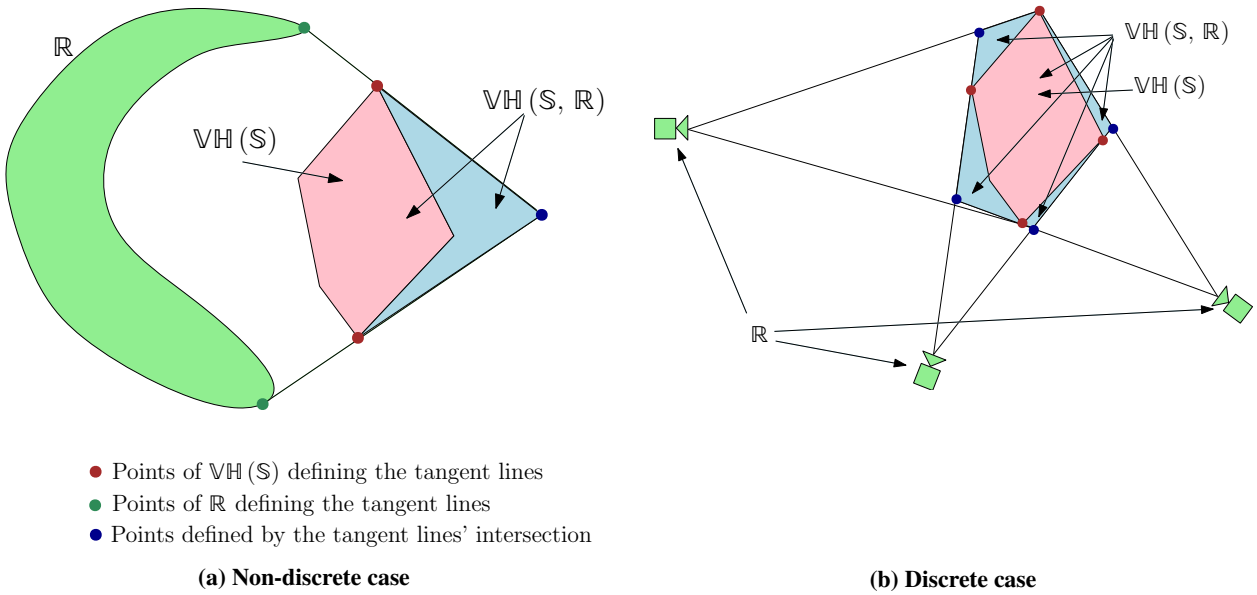


(e) An object $\mathbb{S}'$ with $\mathbb{VH}(\mathbb{S}') = \mathbb{VH}(\mathbb{S})$



(f) The convex hull $\mathbb{CH}(\mathbb{S})$

Fig. 1 An example of the silhouette-based reconstruction (modified and extended from Laurentini [25])



(a) Non-discrete case **(b) Discrete case**

Fig. 2 The construction of the visual hull $\mathbb{VH}(\mathbb{S}, \mathbb{R})$

$\mathbb{VH}(\mathbb{S}) \leq \mathbb{CH}(\mathbb{S})$ where $\mathbb{CH}(\mathbb{S})$ is the convex hull of object $\mathbb{S}$. Interested readers can refer to Laurentini [25] for discussions and hypothesis justification. This implies that the visual hull is bounded by the convex hull, but more information can be recovered from silhouettes. This is because silhouettes contain information about the object concavities. Interested readers can refer to Laurentini [26] for a more detailed discussion. Examples are shown in Figures 1d and 1f.

It is not necessary to have one silhouette per point in the 3D Euclidean space to recover the visual hull. Let $\mathbb{R}_c \in \mathbb{R}^3$ be the subset of the 3D Euclidean space which allows the visual hull to be completely recovered. The only hypothesis on $\mathbb{R}_c$ is that it must enclose completely $\mathbb{S}$ with null interception with the visual hull itself. This is because a camera translating in the direction of its principal axis generates the same viewing cone, thus the same piece of information.

Let $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ be the visual hull of $\mathbb{S}$ obtained from $\mathbb{R}$. $\mathbb{R}$ is a non-discrete subset of $\mathbb{R}_c$ that does not enclose the object $\mathbb{S}$. With reference to the 2D example reported in Figure 2a, $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ is the union of the pink area and the light blue one. The pink area is the visual hull as recovered by $\mathbb{R}_c$, i.e., $\mathbb{VH}(\mathbb{S})$, while the light blue area is an artifact caused by the silhouette-based reconstruction. This artifact exists because $\mathbb{R}$ does not enclose the object $\mathbb{S}$ and is generated by the intersection of the viewing cones. In the 2D example reported in Figure 2a, this compensation is generated by the lines tangent to $\mathbb{R}$ and $\mathbb{VH}(\mathbb{S})$. These lines intersect at a point that helps to define the light blue area. For a three-dimensional case, lines are substituted by planes and intersection points are substituted by intersection segments. By labeling $\mathbb{A}(\mathbb{S}, \mathbb{R})$ this compensation, $\mathbb{VH}(\mathbb{S}, \mathbb{R}) = \mathbb{VH}(\mathbb{S}) \cup \mathbb{A}(\mathbb{S}, \mathbb{R})$. Thus, the visual hull $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ contains $\mathbb{VH}(\mathbb{S})$. Moreover, the visual hull surface $\mathbb{vh}(\mathbb{S}, \mathbb{R})$ coincides with $\mathbb{vh}(\mathbb{S})$ only in the subset of $\mathbb{sa}(\mathbb{S})$ seen from $\mathbb{R}$. In practice, the set $\mathbb{R}$ is discrete and the visual hull $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ is computed from the intersection of a finite number of

viewing cones. This implies that the visual hull is defined by the tangent lines to the object $\mathbb{S}$ and the camera positions. As the camera positions form a discrete set, $\mathbb{vh}(\mathbb{S})$ and $\mathbb{vh}(\mathbb{S}, \mathbb{R})$ coincides only where the lines tangent to $\mathbb{S}$ pass through the discrete points of $\mathbb{R}$, i.e. the camera centers. This situation is depicted in Figure 2b.

As in this work, the silhouettes are modified to compensate for shadows by computing shadow-robust silhouette, the reconstructed visual hull is called hereafter shadow-robust visual hull. The shadow-robust visual hull depends not only on the relative localization between $\mathbb{S}$ and $\mathbb{R}$ but also on the relative geometry among the camera, the light source, and the observed object. This is discussed in details in Section VI.C.

VI. Algorithm Overview

The algorithm is structured in two different parts: the shadow-robust silhouette extraction and the visual hull reconstruction. The main assumptions behind the algorithm are:

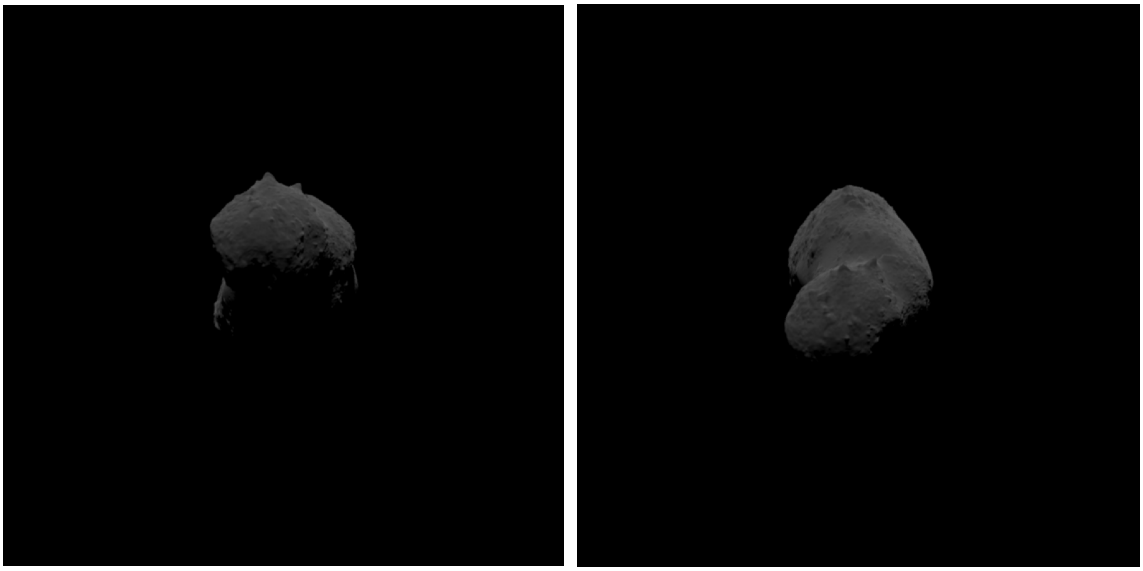
1) Camera is internally calibrated and the intrinsic matrix is known.
2) Spacecraft poses, i.e., position and orientation, with respect to the small body are already been determined. This implies that a localization step has been previously performed to determine the camera's relative poses as in Panicucci et al. [27].
3) The Sun position with respect to the spacecraft is known. The assumption could be relaxed by considering that only the Sun direction is known and the Sun can be placed at infinity.

A. Shadow-Robust Silhouette Extraction

The shadow-robust silhouette extraction is the first important contribution of this work. The silhouettes from raw images are modified to keep only limbs that are surely illuminated by the Sun and thus exploitable for the reconstruction of the visual hull. The main idea is to understand where shadows are present in the image to isolate illuminated limbs. For the sake of completeness, examples of raw Itokawa images are reported in Figures 3a and 3b. These images are used in this section to show the image processing steps.

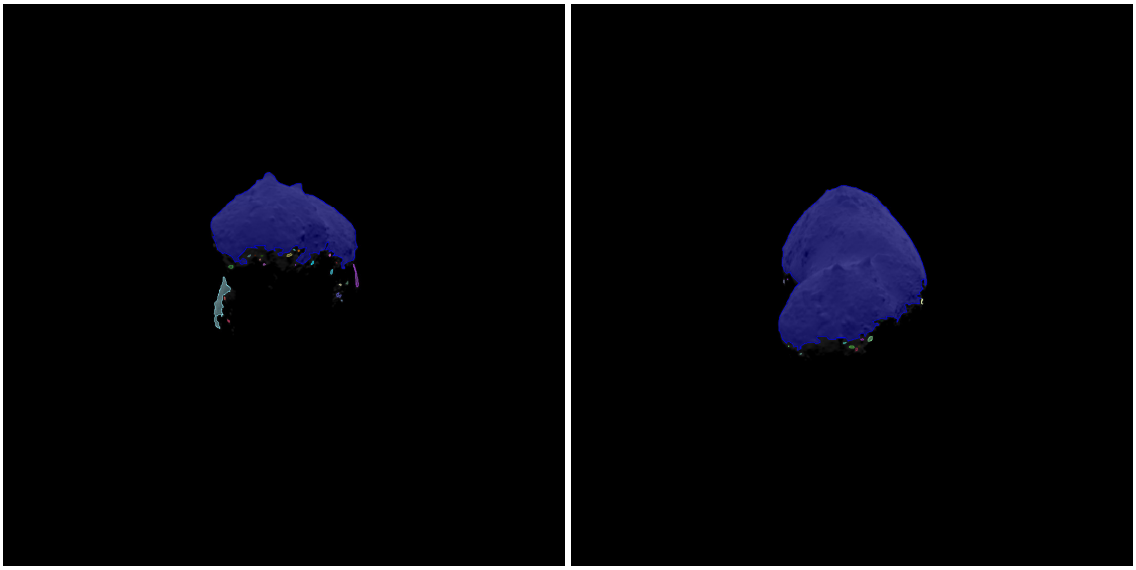
The algorithm starts with the binarization of the image with Otsu's method [28]. The binary map represents the illuminated faces of the small body. The boundary pixels of the binary map form the small body silhouette. Thus, they are extracted to form silhouette edges. Note that, at this step of the algorithm, self-shadowing from a concave shape could generate several contours which belong to the same object, but are perceived as isolated pieces in the image. A polygon is identified per isolated piece as shown in Figures 4a and 4b. These polygons are named initial polygons.

As an initial polygon is generally composed of a high number of vertices, the polygon vertices number is reduced to limit the computational time. The vertices reduction is based on a heuristic that considers the segments length and the



(a) Example 1 (b) Example 2

Fig. 3 Examples of synthetic Itokawa images given to the silhouette-based reconstruction algorithm.



(a) Example 1 (b) Example 2

Fig. 4 Examples of initial polygons extracted from images of Itokawa. Different colors label different initial polygons.

angle between them. The heuristic ψ_i associated with the ith vertex $\boldsymbol{P}_i$ is defined as follows:

$$\psi_i = \left\| \begin{pmatrix} \boldsymbol{P}_{i+1} - \boldsymbol{P}_i \\ 0 \end{pmatrix} \times \begin{pmatrix} \boldsymbol{P}_{i-1} - \boldsymbol{P}_i \\ 0 \end{pmatrix} \right\| \tag{1}$$

where $\boldsymbol{P}_{i-1}$ and $\boldsymbol{P}_{i+1}$ are the neighbors of $\boldsymbol{P}_i$. The heuristic is computed for all the vertices and the vertex associated with the lowest value is removed. The removed vertex neighbors heuristic is updated and the process is repeated until the desired number of vertices is reached. In this paper, the maximal number of vertices is set to 100.

At this step of the algorithm, the visible parts of the small body are identified and their contours are represented as polygons, called initial polygons, whose points are the most relevant ones in terms of length and angle. This ensures that non-redundant long segments are kept to represent the polygon.

Once the initial polygons have been computed, the main issue is that the visible isolated pieces of the small body are not connected to form a unique closed set. And even if they were, part of the small body shape could not be reconstructed as partially shadowed during the observation. To avoid this problem, the information about the Sun position is exploited. The algorithm aims at reconstructing shadows to isolate directly-illuminated limbs. As shadows are caused by the interaction between Sun rays and the unknown small body shape, the basic idea is to reconstruct the shadows directly on the image plane by exploiting the Sun position and the initial polygon, which are the projection of illuminated small body portion.

First, it is worth understanding how shadows are formed. Shadows are defined as the small body surface not impacted by Sun rays. Therefore, shadow boundaries are formed by Sun rays tangent to the small body shape. This is shown in Figure 5 where the Sun, depicted as the orange dot, emits Sun rays depicted with 3D orange and purple lines. The figure is not to scale as the Sun is placed almost at infinity. This implies that Sun rays should be almost parallel lines.

These Sun rays form two types of shadows on a concave shape: self-shadowing and terminator-line shadows. For a convex body, solely the terminator line is present, but it is very improbable to find a purely convex body in the solar system given the high likelihood of finding craterized terrain. On the one hand, self-shadowing boundaries are defined by tangent Sun rays subsequently intersecting the small body after a concavity, as depicted by the purple Sun rays in Figure 5. On the other hand, the terminator line shadows are generated by tangent Sun rays not intersecting the small body after tangency, as depicted by orange Sun rays in Figure 5. The line generated by all these tangent points defines the night-day limit in the small body local time and is also known as the terminator line. Both shadows are present in the image and can not generally be distinguished against the dark background.

The shadows cannot be reconstructed in the 3D world because the small body shape is not known. By knowing how shadows are formed, their projections are computed in the image plane without knowing the 3D real shadows or the 3D small body shape. Note that the proposed methodology does not compute the shadows and projects them, but it computes

directly their projection. Moreover, as these calculations are carried on directly on the image, the overall algorithm is simplified because no complex 3D transformation is required. This yields to compute the so-called shadow-robust silhouette: the union between the initial polygon, which is the projection of an illuminated part of the small body, and the shadow projection generated from the unknown shape. The silhouette is called shadow-robust as the reconstruction computed from these silhouettes is not affected by adverse illumination and shadows. Thus, the intersection of these silhouettes leads to a shadow-robust silhouette-based reconstruction.

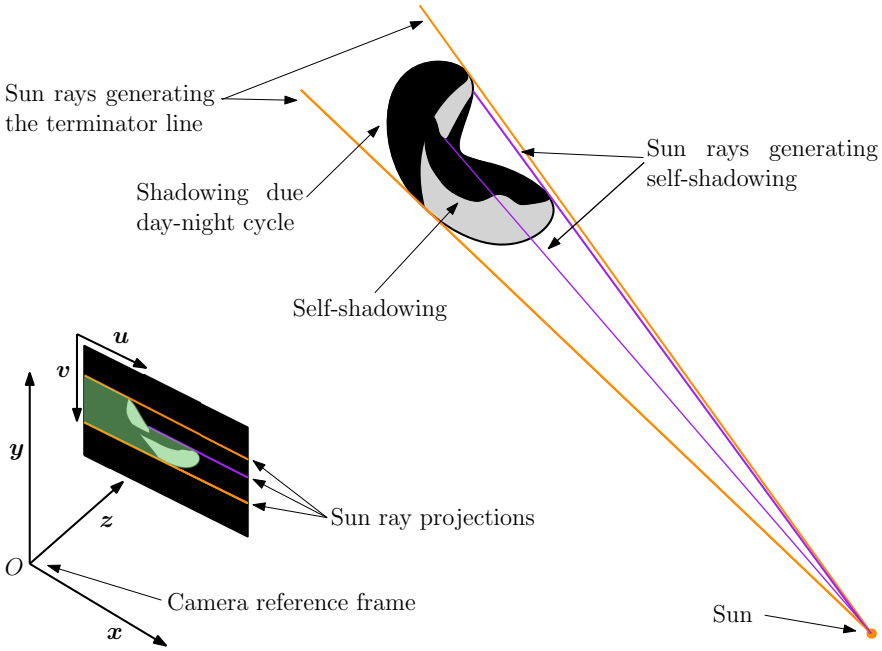


Fig. 5 The geometry between the camera, small body, and the Sun in the 3D space.

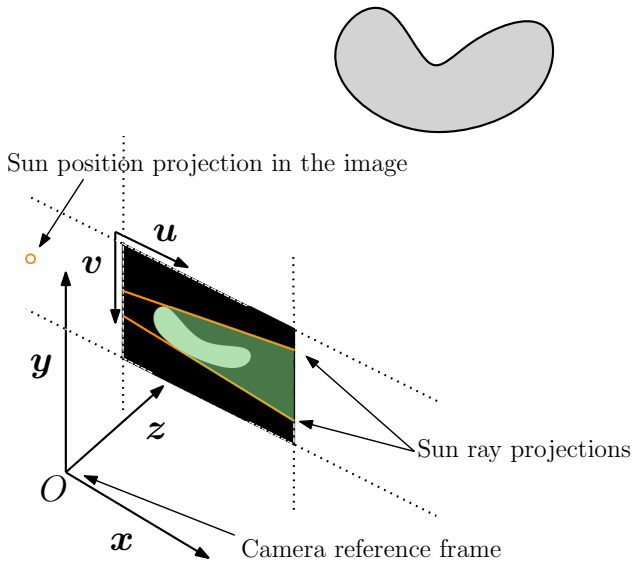


Fig. 6 An example where the shadow-robust silhouette is computed despite absence of shadows.

Note that no shadows may be present in a specific illumination geometry, but the shadow-robust silhouette is computed anyway because only the directly-illuminated limbs are extracted. For example, Figure 6 shows a sketch where no shadows are present as the illumination angle is low, but the shadow-robust silhouette is computed anyway. Indeed, as underlined by Christian [29], the other limbs, i.e., the not-directly-illuminated ones, appear qualitatively correct, but it is not uncommon to have pixelic errors in the terminator line detection. These errors influence the shape reconstruction as its quality depends on the limb detection accuracy. On the one hand, this computation implies that part of the small body, which is not shadowed, is not considered for the visual hull reconstruction as possibly shadowed. On the other hand, only the part of the initial polygon that is directly illuminated is preserved, therefore only sure and exploitable information is processed.

Before analyzing the case of several initial polygons, it is worth considering the case of a single one. First, to compute the shadow projections, the Sun position is projected in the image plane. The 2D lines connecting the Sun position projection with the initial polygon's vertices are gathered. As the initial polygon is the projection of an illuminated part of the body, these 2D lines are the projection of the Sun rays striking the small body in a 3D point whose 2D projection is the polygon vertex. Therefore, the 2D lines tangent to the initial polygon are the 2D projection of the tangent Sun rays generating shadow boundaries. This is sketched in Figure 5 where the 3D Sun rays are colored according to their 2D projection. In particular, the orange Sun rays in Figure 5 are the ones generating the terminator-line shadows. As these tangent Sun rays do not strike the small body after the tangency, their projection is tangent to the initial polygon and intersects the image border. On the contrary, the purple Sun rays in Figure 5 are the ones generating self-shadowing. These tangent Sun rays impact the shape after the tangency, thus their projections are tangent to the initial polygon and intersect the same or another initial polygon.

From these considerations three different sets of points are distinguished:

1) Points of tangency between Sun ray projections and the polygons.
2) Points generated as the intersection between the Sun ray projections and the initial polygons that are on the self-shadowing boundary.
3) Points generated as the intersection between the Sun ray projection and the image borders.

By knowing these points, shadow projections are detected in the image and the initial polygon vertices are removed to account for the shadow projections. Examples of these points sets are reported in Figures 8a.

The shadow projections generated by the polygon are computed through the following steps:

1) The points of tangency are identified by gathering the Sun ray projections tangent to the considered polygon. This is done by computing the local maxima and minima of the distance between the initial polygon vertex and the 2D line connecting the polygon center and the Sun projection. To have a rigorous extraction, this distance is signed thanks to the angle among the Sun projection, the polygon center, and the polygon vertex. This process is depicted in Figure 7. In the figure, the orange line connects the blue initial polygon center and Sun projection.

The distance between the orange line and a vertex is the dotted segment and this distance is signed using the angle among the Sun projection, the polygon center, and the polygon vertex. A positive angle and distance are shown in yellow, while negative ones are in green. In the figure, vertices 3 and 11 are the points of tangency. Other maxima and minima are discarded as the lines from the vertex to the Sun intersect the polygon. Furthermore, examples of the point of tangency computed in this manner are depicted as blue crosses in Figure 8a.

2) The tangent Sun rays projection can intersect the polygon, which implies possible self-shadowing. Several intersections could be computed between a line and a polygon, but only the closest one to the point of tangency is the shadow boundary projection because the Sun ray cannot pass through the small body. If no intersection is found, the projected tangent ray intersects the image borders. Thus, the point of tangency is on the terminator line. This process is depicted in Figure 8a. Note that Sun ray colors are consistent with the ones used in Figure 5. Sun rays generating the terminator-lines shadows are in orange, while Sun rays generating self-shadowing are in purple. In the figures, points generated by intersecting the Sun ray projections are depicted with green crosses.
3) After having computed the points of tangency and the intersection points, the current polygon is modified by removing points covered by the shadow projections. This is performed by merging the initial polygon with a polygon defining the shadow projection. This generates a shadow-robust silhouette.

When more than one initial polygon is present, they are processed sequentially. Initially, the first initial polygon in the list is selected and its shadow projection is computed. An intermediate shadow-robust silhouette is generated and merged with the second initial polygon in the list. The merged polygon is called an intermediate polygon hereafter. Then, the shadow-robust silhouette generated from the intermediate polygon is computed and merged with the following initial polygon. The process is iterated until all the initial polygons are processed. An example of the procedure is reported in red in Figure 8b where the red intermediate shadow-robust silhouette is merged with the magenta initial polygon to obtain an intermediate polygon. The point of tangency is the blue cross, the intersection point is the green cross, and the Sun ray projection is the orange line.

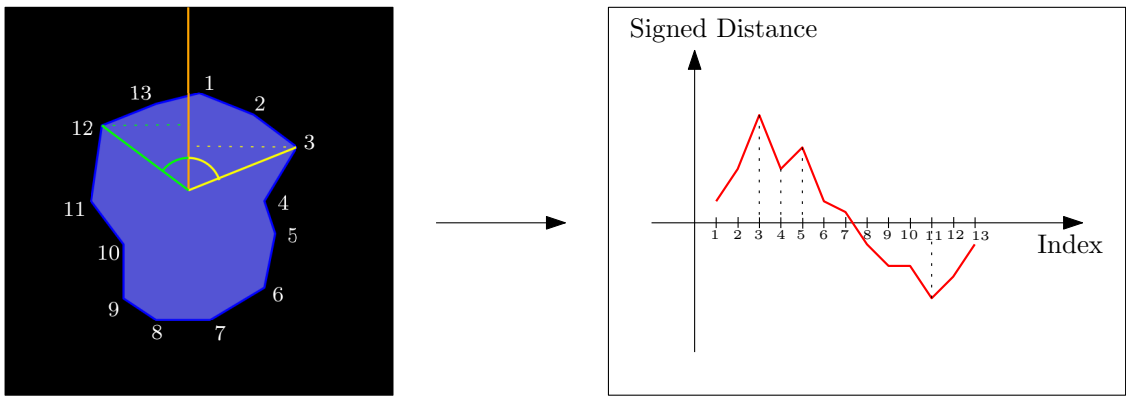
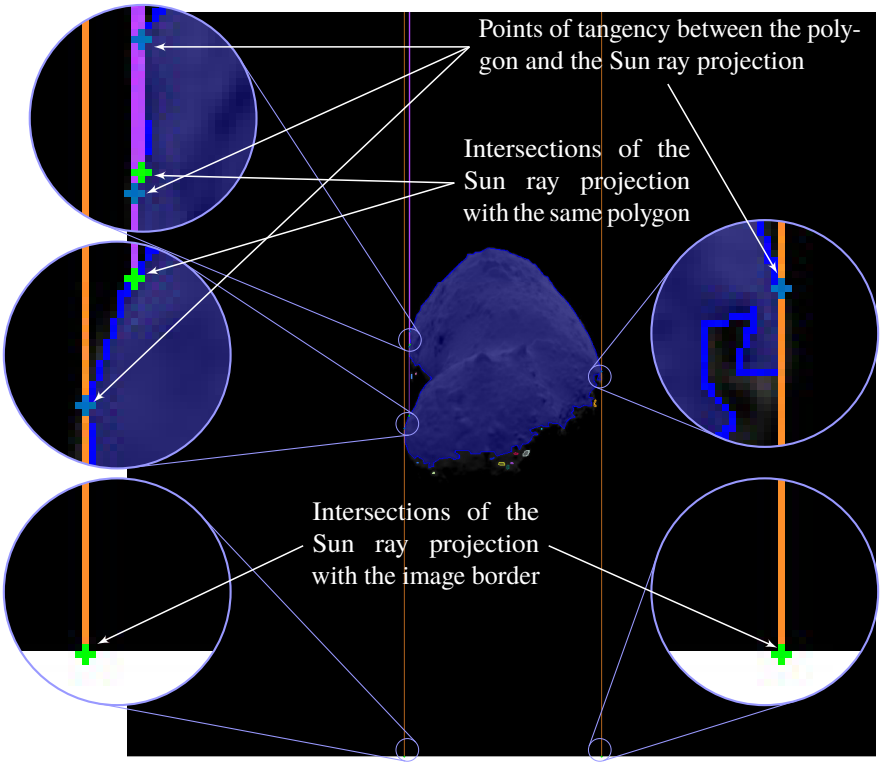
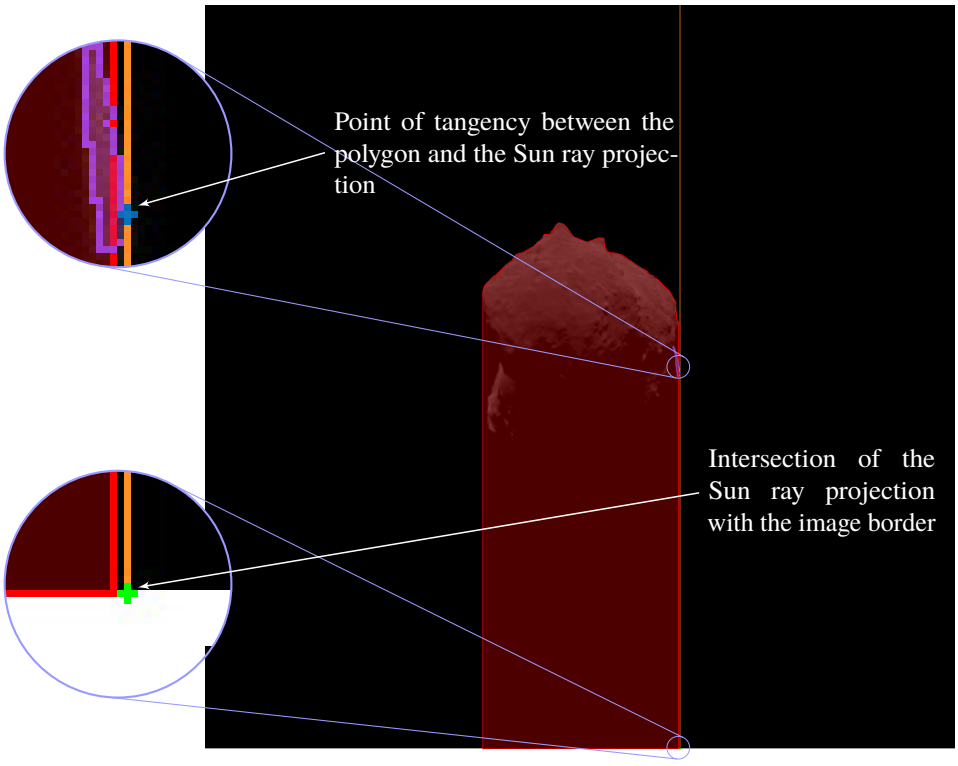


Fig. 7 Points of tangency identification from initial polygon vertices.

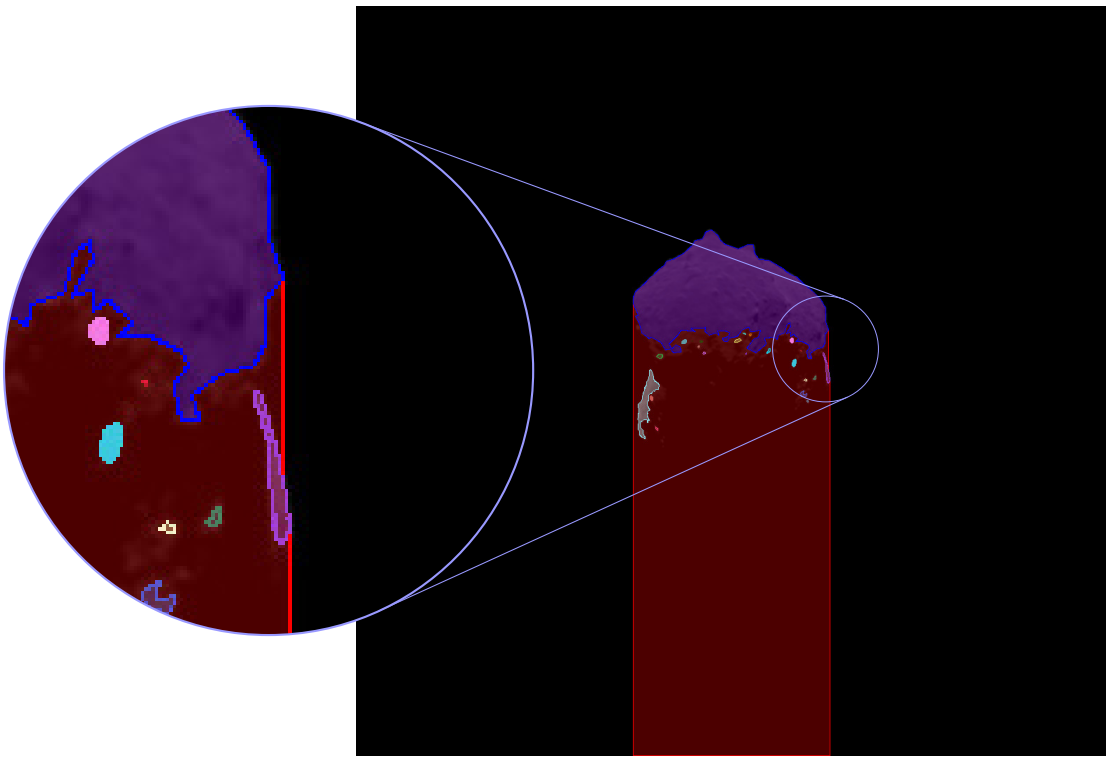


(a) Examples of intersection points and points of tangency.

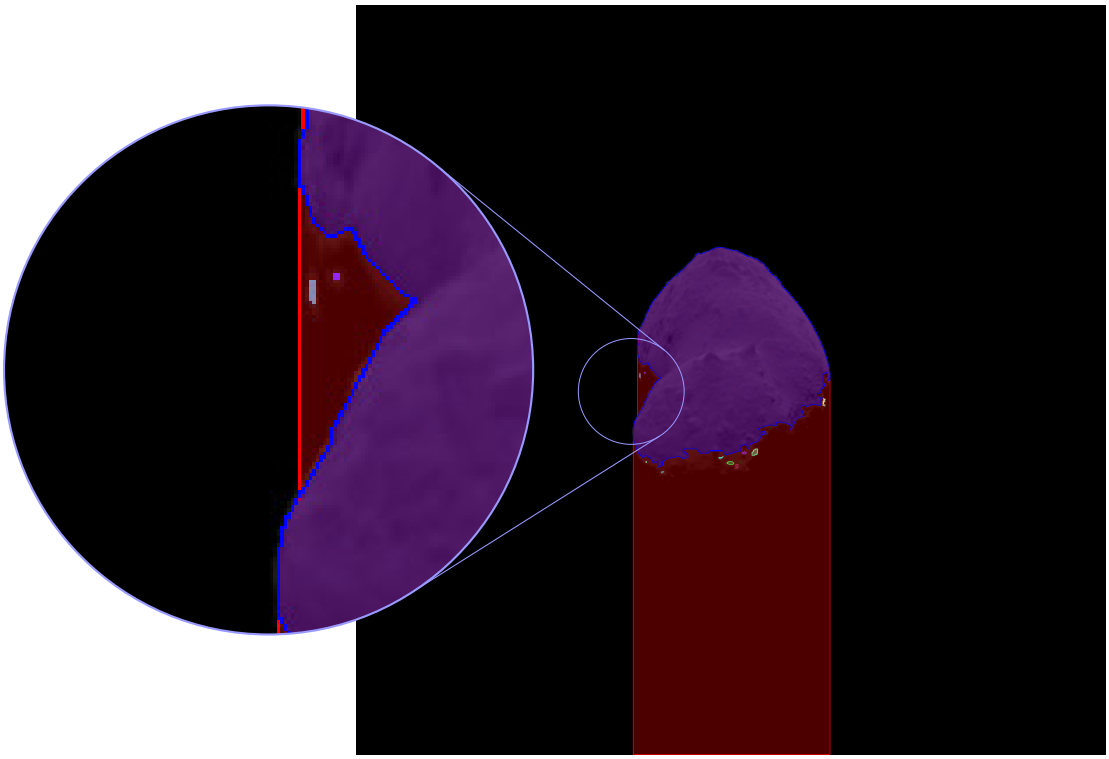


(b) An intermediate shadow-robust silhouette, in red, is merged with the magenta initial polygon.

Fig. 8 Intermediate steps of the shadow-robust silhouette extraction.



(a) In the magnified circle the intersection between the shadows projection and another initial polygon.



(b) In the magnified circle the intersection between the shadows projection and the same initial polygon.

Fig. 9 Examples of shadow-robust silhouettes.

Note that, if the following initial polygon is inside the current intermediate shadow-robust silhouette, the resulting merged polygon is equal to the intermediate shadow-robust silhouette and thus the cycle continues to the following iteration. Figures 9a and 9b report examples of initial polygons covered by the red shadow-robust silhouette generated by the blue initial polygon. In the magnified circles, two examples of possible self-shadowing.

When all the initial polygons are processed, the last shadow-robust silhouette is the final one for the considered image. Examples of the final shadow-robust silhouettes are shown in Figures 9a and 9b where the concave shape of Itokawa causes self-shadowing due to the high illumination angle. The initial polygons, in different colors, are used to generate the shadow-robust silhouette, in red, as explained in the previous steps.

B. 3D Points Computation and Shadow-Robust Shape Reconstruction

Once the shadow-robust silhouettes are extracted from all images, the computation of the 3D points lying on the shadow-robust visual hull is performed. The presented technique is an extension of the work proposed in Boyer and Franco [22] that computes 3D points on the visual hull from a series of silhouettes in a purely geometrical way. This process is chosen as it requires no optimization and can handle a high number of images with limited computational time.

Let $[F]$ be the fundamental matrix that captures the geometry, also called epipolar geometry, between two different views [13]. By knowing the camera intrinsic matrix $[K]$ and the camera pose of the two views, $[F]$ can be computed. Thus, the epipoles, i.e., the projection of one camera center in the other image plane, can be deduced [13]. This geometrical configuration is depicted in Figure 10.

By knowing the camera intrinsic parameters, the ith vertex of the kth shadow-robust silhouette $\boldsymbol{P}_{i,k}$ is backprojected in the 3D space. Let $C = \{c, \boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3\}$ be the camera reference frame. Let $\mathbb{C} = \{C, \boldsymbol{C}_1, \boldsymbol{C}_2\}$ the image frame centered in the image center C. Thus, a ray passing through the camera center $\boldsymbol{r}_{\text{cam}}$ and the point ${}^{C}\boldsymbol{p}_{i,k} = {}^{C}\boldsymbol{r}_{\text{cam}} + {}^{C}\boldsymbol{l}_{\boldsymbol{P}_{i,k}}$ where $\boldsymbol{l}_{\boldsymbol{P}_{i,k}}$ is the line of sight of $\boldsymbol{P}_{i,k}$. this latter is computed as:

$$ {}^{C}\boldsymbol{l}_{\boldsymbol{P}_{i,k}} = [K]^{-1}\, {}^{\mathbb{C}}_{h}\boldsymbol{P}_{i,k} \tag{2} $$

where ${}^{\mathbb{C}}_{h}\boldsymbol{P}_{i,k}$ is the vector ${}^{\mathbb{C}}\boldsymbol{P}_{i,k}$ in homogeneous coordinates.

The same procedure can be performed for the $(i+1)$th vertex of the kth shadow-robust silhouette $\boldsymbol{P}_{i+1,k}$. By backprojecting both the $(i+1)$th vertex and the ith vertex, the ith segment of the kth shadow-robust silhouette is backprojected in the 3D world to form a plane portion defined by the two rays. This process is represented in step (1) in Figure 10 where the brown circles in the left camera view are backprojected in the 3D space. The backprojections form the green rays defining the light green plane portion that is the backprojection of the segment defined by the ith and $(i+1)$th vertices. Moreover, these rays are tangent to the small body at the empty brown squares. Recall that the empty

brown circles are not known during shape reconstruction as the shape is not determined yet.

The obtained plane portion can be projected in a second view thanks to epipolar geometry and generates a 2D polygon. This polygon is finally intersected with the shadow robust silhouette in the second view to obtain a 3D plane portion that belongs to both silhouettes. Note that four geometrical entities are defined:

1) The 1D segment in the first view defined by the ith and $(i+1)$th vertices.
2) The 3D plane portion which is the backprojection of the 1D segment. This implies that a 3D point in the plane portion is projected in the first camera view as a point of the 1D segment. Analogously, the backprojection of a point in the 1D segment forms a ray belonging to the 3D plane portion.
3) The 2D polygon in the second view computed by projecting the 3D plane portion in the image. This 2D polygon is obtained by intersecting the plane portion projection with the image plane. This 2D polygon is called hereafter plane portion projection. Examples of this procedure are reported in Figures 11a - 11d.
4) The 2D polygon in the second view computed by intersecting the plane portion projection and the second view shadow-robust silhouette. This 2D polygon is called hereafter intersection polygon. The intersection between the plane portion projection and the silhouette border is done through brute force algorithm by iterating over all segments. A logarithmic search could be achieved with a segment pre-ordering step as in Matusik et al. [30]. Examples of this procedure are reported in Figures 11a - 11d.

Figures 11a - 11d show examples of plane portion projections and intersecting polygons. Figure 11a depicts an example where both 1D segment vertices epipolar lines intersect the image plane. This implies that both vertices are visible in the second camera view. Thus the whole 1D segment could be observed in the image. Moreover, as the plane projection has no null intersection with the second view silhouette, the segment is observed. Note that in this case the whole 1D

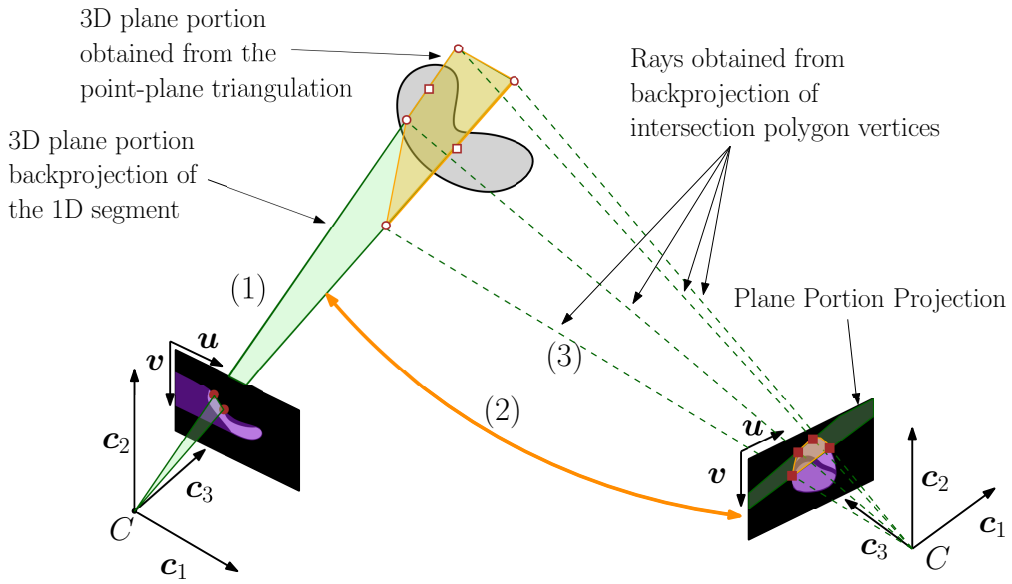
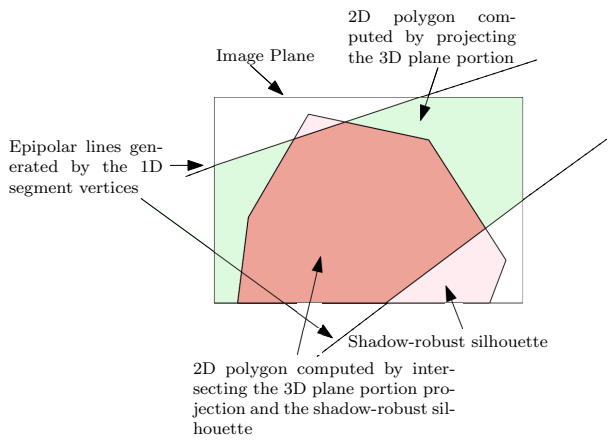
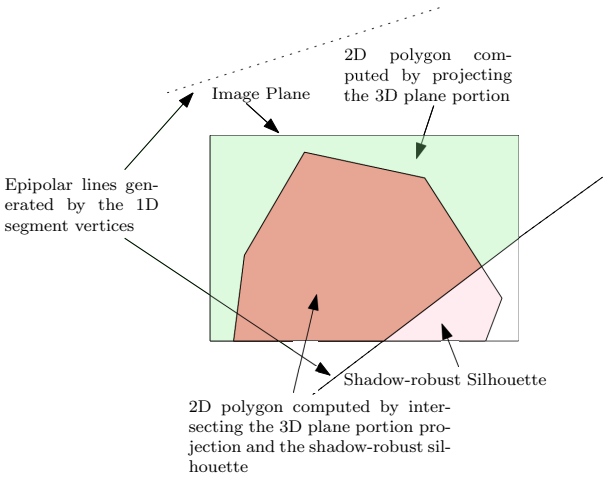


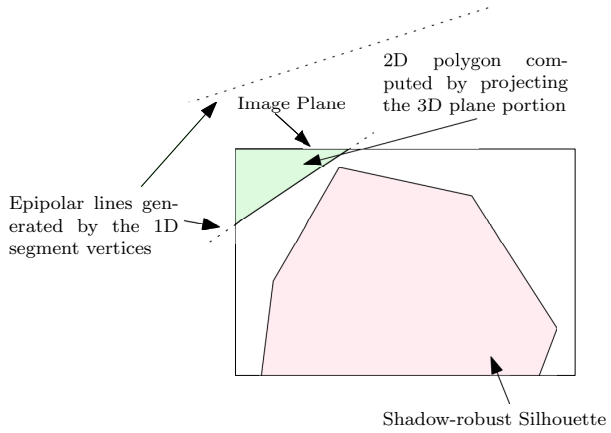
Fig. 10 The 3D point computation with 2 views. Steps (1), (2), and (3) are detailed in the text.



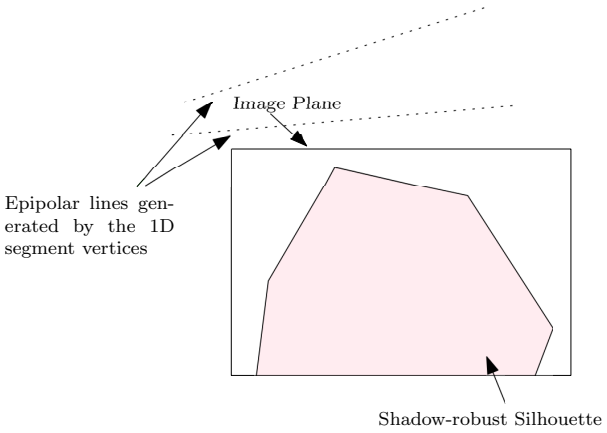
(a) Both epipolar lines generated the image plane.



(b) Only one of the epipolar lines intersect the image plane.



(c) No intersection between the shadow-robust silhouette and the 3D plane portion projection.



(d) No intersection between the epipolar lines and the image plane.

Fig. 11 An example of the manipulations of the plane portion in the second view.

segment is observed as its vertices epipolar lines intersect the silhouette. Figure 11b shows a case where only one 1D segment vertices epipolar line intersects the image plane. This means that only a portion of the 1D segment could be observed in the second view. Moreover, the silhouette and the plane portion projection have no null intersection, but the segment is not visible in its whole. Figure 11c illustrate a case where only one 1D segment vertices epipolar line intersects the image plane, but no intersection exists with the silhouette. This means that the 1D segment could be visible in the image, but the intersection with the silhouette states that it is not. Figure 11d depicts an example where no intersection between the plane portion projection and the image plane exists. This means that the 1D segment is not visible by the second view.

If the cases presented in Figures 11c and 11d happen, the 1D segment and its plane portion are discarded. This is because the considered first view's silhouette segment does not belong to the visual hull. Otherwise, the intersection polygons are the projection of a series of 3D polygons belonging to the 3D plane portion. Thus, the 3D polygons are the

backprojections of the 2D polygons formed by the intersection between the plane portion projection and the second view silhouette. Note that the 3D polygon vertices are projected in the first camera view as 2D points of the 1D segment. This happens because the 3D plane portion is the backprojection of the 1D segment. These relationships are depicted in step (2) in Figure 10 where the light green plane portion is projected as the light green 2D polygon in the right camera view. The light green plane portion projection is intersected with the second view silhouette, in purple. The intersection polygon is reported in orange whose vertices are the four full brown squares. Figure 10 shows only four points and one polygon generated in this way. Nevertheless, the intersection between the plane portion projection and the silhouette can generate an arbitrary number of points and an arbitrary number of polygons as two arbitrary 2D polygons are intersected. Examples of intersection are reported in Figure 11a and 11b for simple geometries.

The 3D polygon vertices can be triangulated, as depicted in step (3) in Figure 10. This triangulation is computed by intersecting the 3D plane portion and the 3D rays obtained by backprojecting 2D intersection polygons vertices from the second view. Note that a solution must exist because the intersection between the plane portion projection and the second view's silhouette is not null. It is worth noting that, because of this geometrical construction, no optimization is required as the 3D triangulation is exact. The triangulation is performed with the algorithm reported in Appendix A.

Similarly, the computed 3D polygons are projected in a third view, as shown in Figure 12. Then these polygon projections are intersected with the third view silhouette to find new 2D intersection polygons to be triangulated with the plane portion. This helps to find 3D polygons that belong to the three silhouettes simultaneously. In Figure 12, the orange 3D polygon is projected in the third view and intersected with the associated silhouette. The intersection polygon vertices are backprojected in the 3D space as rays that intersect the 3D plane portion generated from the first view. This generates a 3D polygon that belongs to the three silhouettes simultaneously.

The process can be generalized to an arbitrary number of views and the output is a 3D polygon set generated from one silhouette and belonging to all the other silhouettes.

To compute all the points lying on the shadow-robust visual hull, all the shadow-robust silhouettes are backprojected in the 3D space and intersected with all the other shadow-robust silhouettes. This process generates a set of 3D points - the 3D polygon vertices - which belong to all the shadow-robust silhouettes at the same time and, thus, lie on the shadow-robust visual hull of the observed small body. Finally, 3D points are triangulated to find a polyhedral visual hull. The surface reconstruction is based on the Ball-Pivoting algorithm [31].

C. Discussion on the algorithm

The visual hull is introduced and formally defined in a general framework in Section V. In this section, its estimation is discussed in the context of small-body approach. The considered parameters are (see Figure 13 for definition):

1) The approach angle which is the angle between the small-body rotation axis and the approach direction, i.e., the camera-small-body direction.

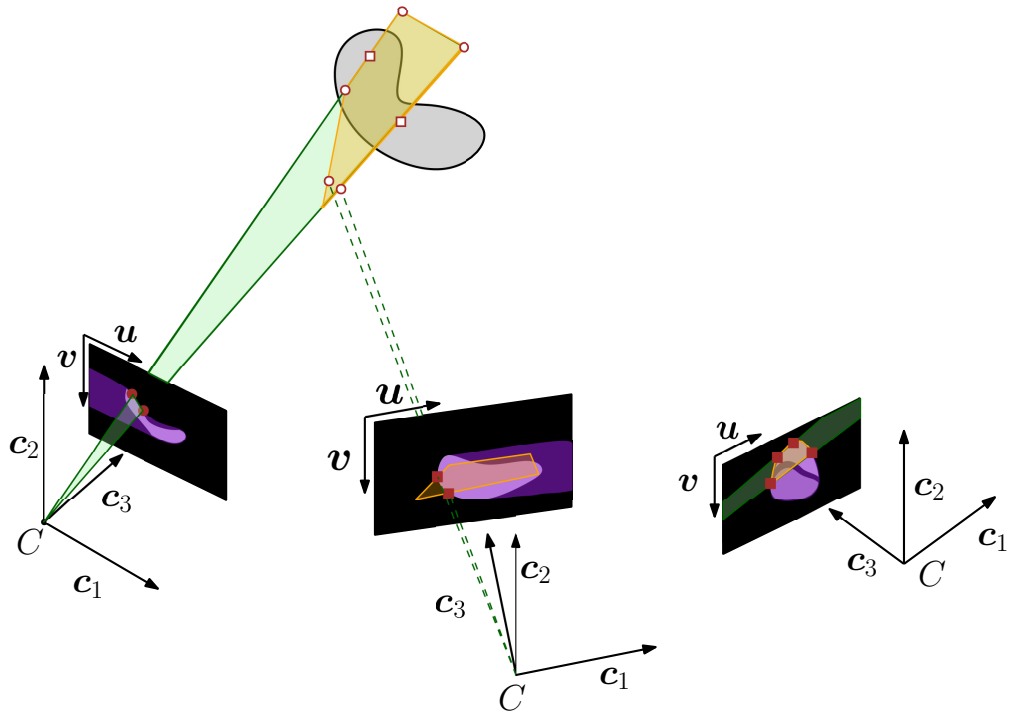


Fig. 12 The effect of adding a third view in the 3D point computation from two views.

2) The illumination angle which is defined as the angle between the approach direction and the Sun-small-body direction.

3) The obliquity, also known as the axial tilt, which is the angle between the small-body rotation axis and the small-body orbital plane.

First, the approach angle effect is analyzed. It is worth noting that the relative dynamics between the camera and the small body is decomposed into two components: the spacecraft's motion with respect to the small body in the inertial reference frame and the motion due to the rotational dynamics of the small body. The rotational motion is the component that enables observing different faces of the small body to bound the visual hull. Indeed, the inertial approach is usually dynamically slower and it heads in the direction of the small body. As a consequence, the approach angle is a central parameter to reconstruct the visual hull. It is considered in the following analysis that the illumination is perfect and the rotation axis is inertially-fixed.

Let the small body north hemisphere be the hemisphere in the positive direction of the rotation axis and the small body south hemisphere be the one in the opposite direction. If the approach direction and the rotation axis are not perpendicular, the body is partially never observed by the camera during the approach. Note that this is a comment purely based on geometrical consideration, without considering any illumination effect. In particular, if the angle is less than 90° the southern hemisphere is not observed, otherwise, the northern hemisphere is not visible. These situations cause the algorithm to compensate for the non-observability of a shape portion by conservatively reconstructing

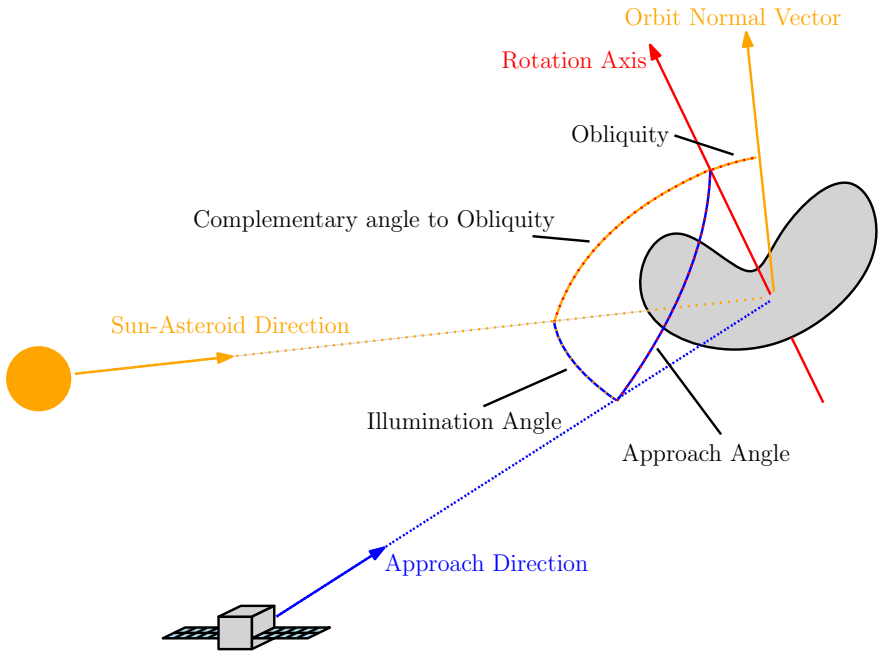
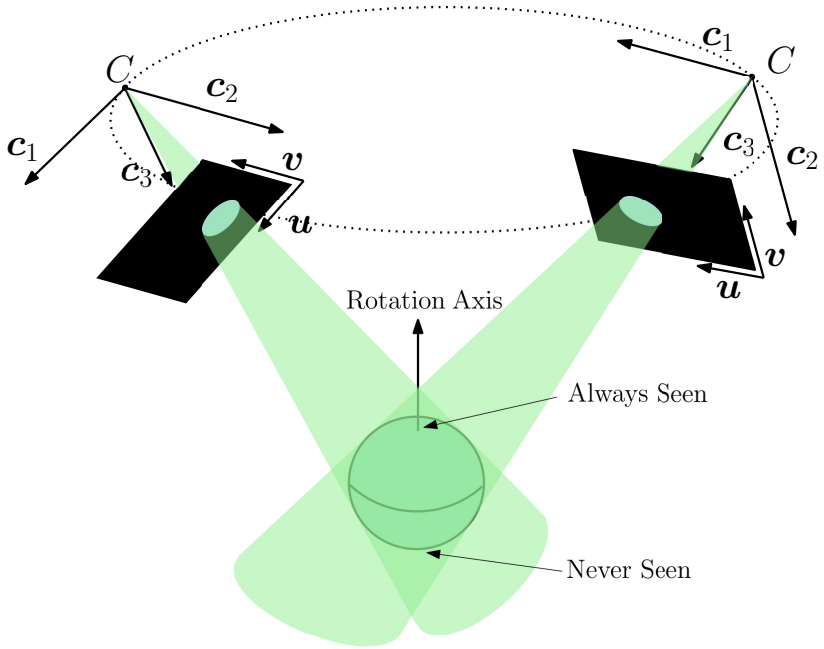


Fig. 13 The definition of the approach angle, the illumination angle, and the obliquity.

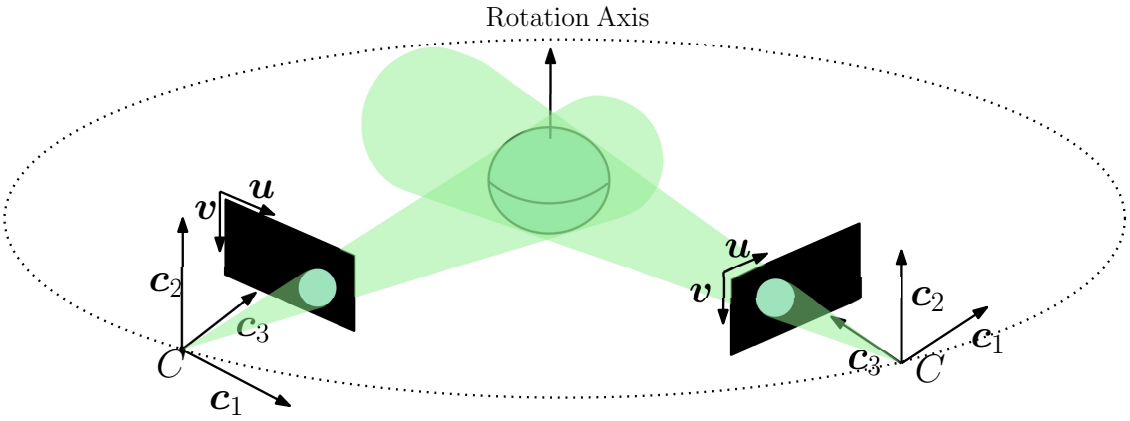
the visual hull. Note that this compensation is not present only in the not-observed hemisphere but also in the constantly-observed one. They are both generated by the tangent cones to the shape. In particular, it is important to note that the constantly-observed hemisphere cannot be reconstructed. Indeed, it never appears in the silhouettes as always lying inside the silhouettes and never belonging to any limb.

This situation is depicted in Figure 14a where the relative localization of two cameras is shown with respect to a sphere rotating about its rotation axis. Recall that $\mathbb{VH}(\mathbb{S}) = \mathbb{S}$ for a sphere. The visual hull $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ is the intersection of the two viewing cones in green. The difference between $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ and $\mathbb{VH}(\mathbb{S})$ can be reduced in the equatorial region by adding more views. This is not possible in the southern and northern hemispheres where the reconstruction is compensated due to the hemispheres' unobservability. Alternatively, when the spacecraft approaches the small body perpendicularly to its rotation axis, the rotational dynamic allows the north and the south hemispheres to be observed almost completely and, with perfect illumination, the visual hull $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ shows no compensation with respect to $\mathbb{VH}(\mathbb{S})$. In this case, the differences between $\mathbb{VH}(\mathbb{S}, \mathbb{R})$ and $\mathbb{VH}(\mathbb{S})$ are caused by the discretization of the observation space which can be limited by adding images. This is shown in Figure 14b where the dotted line is the trajectory induced by the rotational dynamics.

Second, the influence of the illumination angle is studied. The illumination angle influences how many observed faces are in shadow in the image. The more the illumination angle increases, the more shadows are present in the image. As previously stated, these shadows are due not only to the local day-night line but also to self-shadowing. An important contribution of the presented algorithm is the shadow-robust silhouette extraction. The shadow-robust silhouette



(a) The geometry when the approaching direction is not perpendicular to the rotation axis.



(b) The geometry when the approaching direction is perpendicular to the rotation axis.

Fig. 14 The influence of the approach angle on the visual hull reconstruction.

computation ensures that only the illuminated limbs are considered. But it also avoids storing in the shadow-robust silhouette limbs of the small body which could be self-shadowed. As a consequence, some concavities, which are not surely exposed to the Sun light due to self-shadowing, are not reconstructed. These concavities are the ones lying on the plane defined by the Sun-small-body direction and the small body rotation axis at the silhouette's time. An example is shown in Figure 9b where the concavity in the asteroid neck is excluded from the silhouette as it is not possible to determine whether the black region is a shadow or the background. This is due to the impossibility to guarantee with a method purely based on the image plane when the object $\mathbb{S}$, which is not known, is self-shadowed.

Finally, an important factor that affects the shadow-robust visual hull reconstruction is the small body obliquity. This

parameter defines seasonal effects, thus observability during the approach of the southern and northern hemispheres. In the most favorable case, the obliquity is null and the spacecraft observes all the small body faces. This implies the absence of seasonal effects due to the orbit of the small body around the Sun. The northern and the southern hemisphere are equally illuminated during the small-body year. On the contrary, if the obliquity is not null, the observability of one of the hemispheres depends on the small-body season at approach time. Let summer be the period of the small body orbit when the north pole is oriented toward the Sun. The other small body seasons are defined according to the small body orbital motion following the Earth seasons convention. It is important to note that the approach is considered at fixed-season as the orbital motion is slow when compared to the approach duration. Moreover, as the rotation axis orientation is not known before arrival, it is unpredictable to say whether the approach takes place when the northern hemisphere in summer or winter. Thus it is unpredictable to say which hemisphere is the shadowed one. The hemisphere which is not directed toward the Sun is not observable. The shadow-robust silhouette computation compensates for this unobservability by extending the extremal points of the terminator line until the image border as in Figures 9a and 9b. Then, when processing these shadow-robust silhouettes, the visual hull computation compensates for the non-observability of the shadowed hemisphere. Note that this is not due to the shadow robust implementation but to the fact that silhouette-based reconstruction methods aim at reconstructing the visual hull and not the object shape.

VII. Numerical Results

A. Test Cases Description

Numerical simulations reproducing the approach phase to a small body are studied. The spacecraft is in close approach with the small body which implies that the probe is not under the influence of the small body gravity field. As the approach time is much smaller than the small body orbiting period around the Sun, the small body orbital reference frame is considered inertial. Thus equations of motion are integrated in this frame. It is worth noting that the spacecraft is orbiting the Sun, but the swept conic arc is so short that it can be approximated by a straight line, so as to ignore the Sun's gravity.

Moreover, the small body rotates around an inertially-fixed axis with constant angular velocity. The spacecraft orientation in the inertial frame is considered known and the camera is pointed to the center of the small body.

The probe is placed at an approaching distance from the small body and slowly approaches the small body. The mapping camera, like Hayabusa's AMICA [32] or OSIRIS-REx's MapCam [33], is constantly observing the small body. Camera parameters are reported in Tables 1 for sake of completeness. Images are simulated with the SurRender® software [34]. Small-body shapes, Bennu and Itokawa in this study, are taken from the Planetary Data System - Small Body Node by downloading the highest resolution polyhedral shape model. The shape model is given to SurRender® by providing the small-body bulk optical properties, like albedo, reflectivity, and diffusivity, and assuming the Hapke

BRDF (Bidirectional Reflectance Distribution Function) [35] to obtain high-fidelity simulated images. The texture is added to the small body to take into account the small-body surface albedo. The camera optics is represented with a Gaussian Point Spread Function (PSF) and a pin-hole projection model. Numerical values, including 3D meshes, are taken from operational missions scenario [32, 33, 36–38].

Five test cases are analyzed hereafter. First the shape reconstruction for the asteroids Bennu et Itokawa is studied under perfect observational conditions. Second, the algorithm behavior without perfect observability is studied in two different cases. On the one hand, in the "Bennu without perfect observability" test case the rotation axis is tilted to point toward the spacecraft's approach direction. This underlines the influence of the approach angle when the observational geometry does not guarantee perfect observability. On the other hand, in the "Itokawa without perfect observability" test case the rotation axis is tilted to point toward the Sun-small-body direction. This underlines the influence of the illumination angle when seasonal effects are present. For the sake of completeness, the illumination angle, the obliquity, and approach angles are reported in Table 2 for all simulated cases. Examples of synthetic images are shown in Figures 15a and 15b.

Finally, the algorithm is tested over Hayabusa's mission images by gathering the geometry from the mission kernels. This highlights the algorithm behavior including pose errors and real camera errors.

B. Bennu test case

The first simulation studies the shape reconstruction of Bennu, an asteroid of 246 m mean radius which is the archetype of a convex object. The spacecraft is slowly approaching the asteroid and observing it for one rotational period taking one image every 21.2°. The illumination angle, the obliquity, and the approach angles are reported in Table 2. The

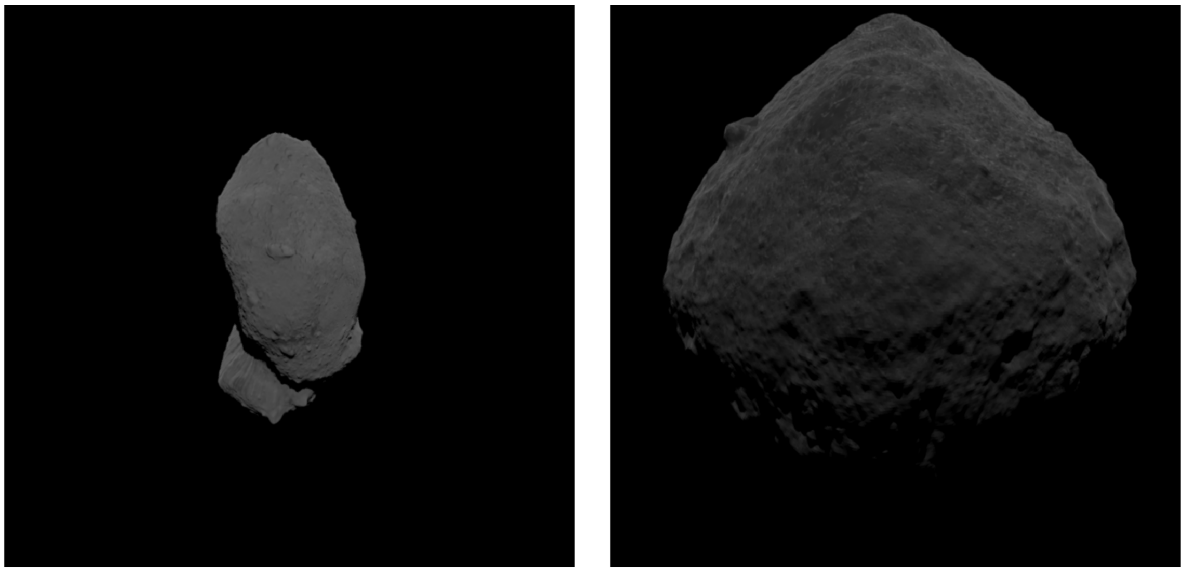
Table 1 Numerical values for the camera parameters.

Camera	FoV [1] [°]	f [2][mm]	$s_x = s_y$ [3][μm]	Image Size [px]	$\boldsymbol{C}$ [4][px]
OSIRIS-REx's MapCam	3.99	125	8.5	1024×1024	$[512, 512]^T$
Hayabusa's AMICA	5.8	121	12	1024×1024	$[512, 512]^T$

[1] FoV is the camera field of view
[2] f is the camera focal length
[3] s_x and s_y are the pixel physical size in x and y components
[4] $\boldsymbol{C}$ is the camera center

Table 2 Angles for the simulated test cases.

Test-case	Obliquity [°]	Approach Angle [°]	Illumination Angle [°]
Bennu with perfect observability	0	0	54
Itokawa with perfect observability	180	90	12
Bennu without perfect observability	123.82	150	54
Itokawa without perfect observability	150	100.82	77.42



(a) Itokawa test-case **(b) Bennu test-case**

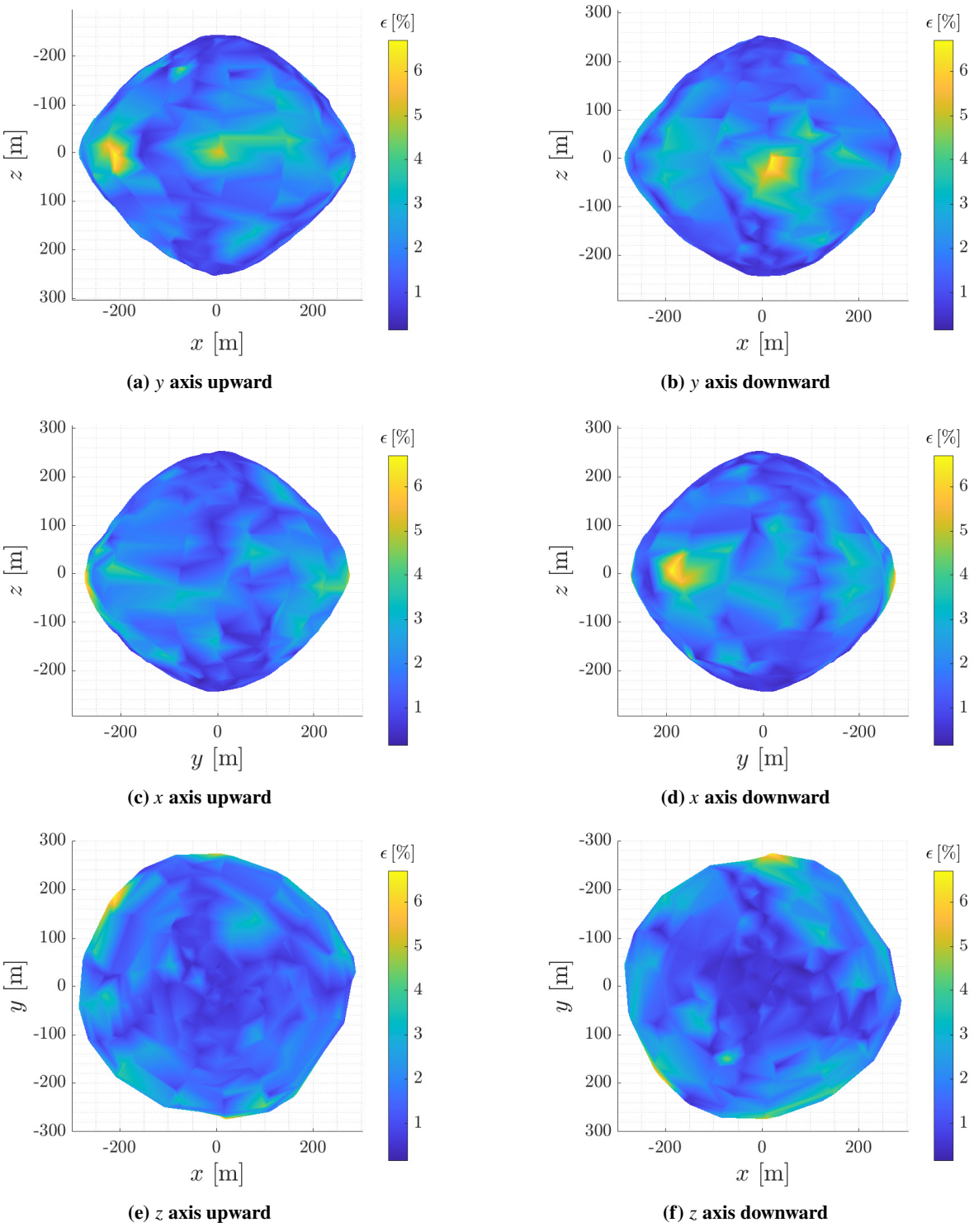
Fig. 15 Examples of synthetic images for the Bennu and Itokawa test case with perfect observability.

Table 3 Performance indices for the analyzed test cases. R_{ast} is the small body mean radius.

Test-case	$\epsilon_{\max}$ [%]	ϵ_{mean} [%]	σ_ϵ [%]	δV [%]	# Images	R_{ast} [m]
Bennu with perfect observability	7.15	1.53	1.13	4.47	17	246
Itokawa with perfect observability	30.74	1.63	7.22	18.42	48	165
Itokawa with mission data	26.35	2.2	7.98	15.21	215	165

high value of the illumination angle implies that the body is partially visible and the proposed conservative silhouette extraction algorithm compensates efficiently this behavior. As stated in Section VII.A, the absence of reconstruction compensation is achieved as the rotation axis is perpendicular to the asteroid orbital plane.

To evaluate the quality of the reconstruction the normalized Hausdorff distance ϵ is used as a performance index. The Hausdorff distance of a vertex of the reconstructed shape is defined as the minimal distance between this vertex and the vertices of the reference small-body polyhedron. The obtained value is then normalized with respect to the small-body mean radius. The normalized Hausdorff distance ϵ between the reconstructed visual hull and the reference shape is depicted in Figure 16a - 16f on the reconstructed visual hull, while a more compact representation is given in Figure 17 by showing the PDF (Probability Density Function). The reconstruction, shown in Figures 18, is well performed with a mean error of 1.53%, i.e. $\sim$ 3.76 m, and a variance of 1.13%, i.e. $\sim$ 17.6 m (see Table 3). The mean error and the associated variance are low which imply good reconstruction. The reconstruction is considered well performed as meter precision accuracy reconstruction is obtained from an approach distance of 8.6 km. Moreover, the maximal error is 7.15%, i.e. $\sim$ 17.6 m. This is possible because the conservative visual hull is close to the real shape and because the approach angle and obliquity have a favorable configuration. Moreover, the volume is well estimated with an error of



(a) y **axis upward**

(b) y **axis downward**

(c) x **axis upward**

(d) x **axis downward**

(e) z **axis upward**

(f) z **axis downward**

Fig. 16 The reconstruction error ϵ projected on the different Cartesian planes for the Bennu test case

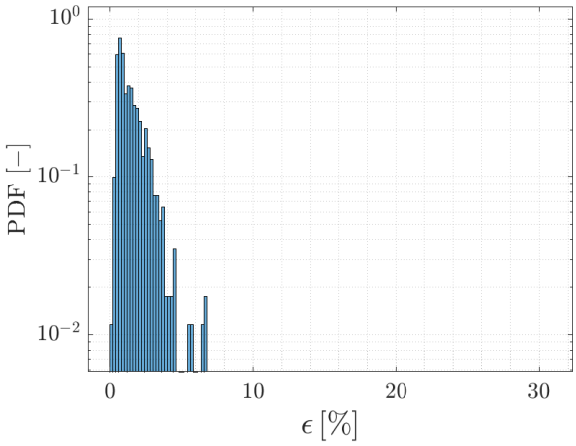


Fig. 17 The histogram of the reconstruction error ϵ for the Bennu test case.

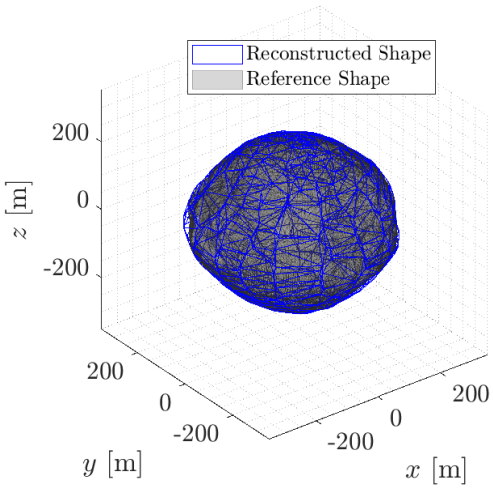


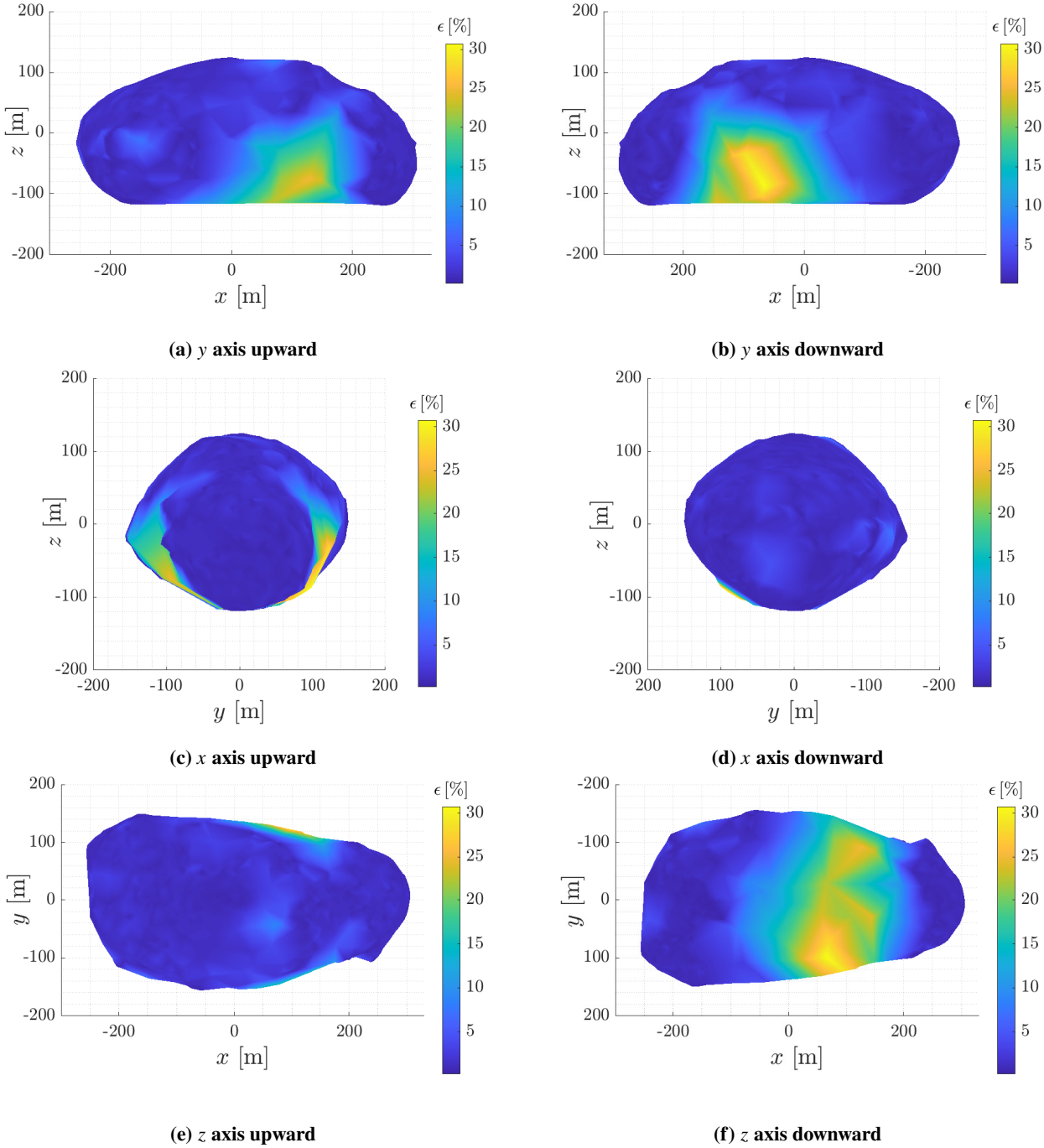
Fig. 18 Reconstructed visual hull for the Bennu test case

4.47% (see Table 3) which can be improved by adding images from other phase angles to constrain the object visual hull.

C. Itokawa test case

The second simulation considers the reconstruction of asteroid Itokawa, a concave object with a 165 m mean radius. The spacecraft takes images every 7.5° for one rotational period, extracts the conservative silhouette, and reconstructs the shape. This test case is chosen as the shape of Itokawa is concave and exhibits a high degree of self-shadowing when the illumination angle grows (see Figures 4a and 4b). The illumination angle, the obliquity, and the approach angles are reported in Table 2.

A detailed map of the error between the reference shape and the reconstructed visual hull is shown in Figures 19a - 19f, whereas in Figure 20 the numerical PDF of the error ϵ is shown. The reconstruction, shown in Figure 21, agrees on the two lobes with high accuracy but some concavities are not well reconstructed. As discussed in Section VI.C, these concavities are not detected because of two reasons: either they do not belong to the visual hull $\mathbb{V}(\mathbb{S}, \mathbb{R})$ or the shadow-robust silhouettes extraction impedes their detection as not surely illuminated. This latter case happens in the north and south poles' concavities given the approach geometry. Indeed, the rotation axis is directed as the z axis. Thus, the approach trajectory and the illumination direction lie on the x-y plane. This is the reason why high values of the error ϵ are concentrated in the south pole. Moreover, this is consistent with the extended tail of the PDF in Figure 20. The relative volumetric error δV (see Table 3) is relatively high, i.e. 18.42%, because of the same reason. Note that the reconstruction under perfect illumination, not reported in this work for the sake of shortness, detects well the north and south poles concavities, as they belong to the visual hull under perfect illumination. Nevertheless, a relative error on the volume δV of 13.05% is present in accord with Baker and McMahon [23]. This is because the equatorial concavities



(a) y **axis upward**

(b) y **axis downward**

(c) x **axis upward**

(d) x **axis downward**

(e) z **axis upward**

(f) z **axis downward**

Fig. 19 The reconstruction error ϵ projected on the different Cartesian planes for the Itokawa test case

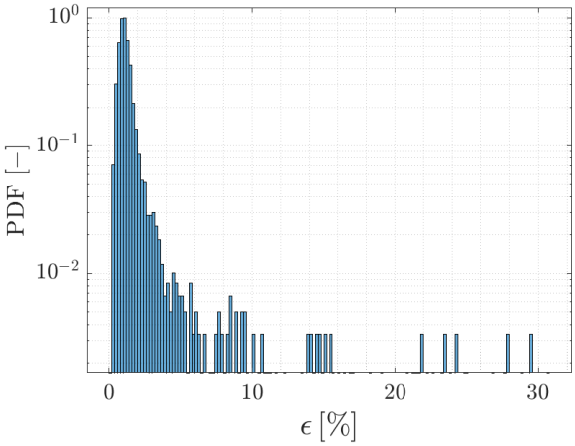


Fig. 20 The histogram of the reconstruction error ϵ for the Itokawa test case.

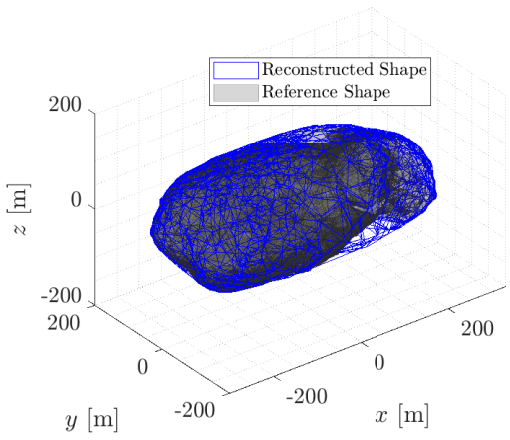


Fig. 21 Reconstructed visual hull for the Itokawa test case

are not captured as never belonging to the small-body silhouettes. The histogram of the error ϵ is less concentrated near zero which implies a worst visual hull reconstruction when compared to Bennu (see Table 3 for error statistics).

D. Effects of non-perfect observability

In previous simulations, the obliquity and the approach angle are chosen to provide a visual hull reconstruction not affected by seasonal and geometrical artifacts. In this subsection, two simulations are shown to analyze the compensation generated from the algorithm during the reconstruction.

First, the influence of the approach angle is studied by keeping the same setting as the Bennu test case and by tilting the rotation axis with respect to the approach direction. The approach angle is 150°, thus the south pole is constantly observed and never belongs to the small body silhouette. On the contrary, the north pole is never observed. The visual hull compensates for this non-observability as shown in Figure 22. Indeed, the viewing cones' intersection does not allow points to be reconstructed when they do not belong to the silhouettes. This implies that the two "mountains" are reconstructed around the rotation axis direction, in red in Figure 22. The number of images is equal to the previous simulation as reported in Table 3. The equatorial region reconstruction is well performed as it belongs to the silhouette limbs, whereas the poles are substituted with the two "mountains", i.e., the viewing cones' intersections. Despite these compensations computed by the algorithm, it is worth noting that the reconstructed shape generates silhouettes consistent with the ones given to the algorithm.

Second, the effect of adverse obliquity is studied with synthetic images of asteroid Itokawa by tilting the rotation axis in the Sun-asteroid direction. The obliquity is set to 150°, the illumination angle is 77.42°, and the approach angle is 100.82°. The value of the obliquity implies that the south pole is never observed as the hemisphere is wintered. The value of the illumination angle has been increased with respect to the previous Itokawa test case to worsen the seasonal

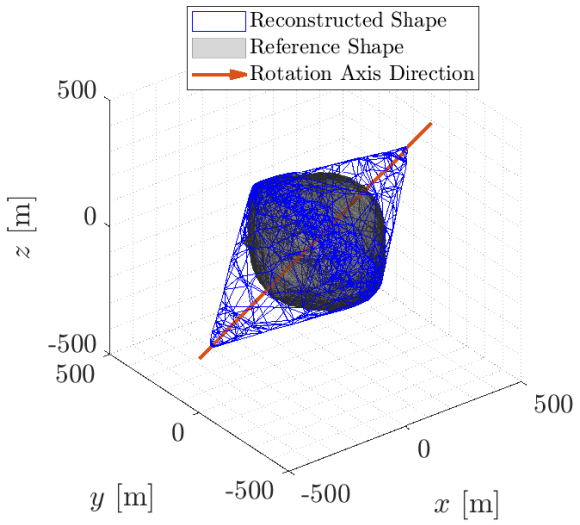


Fig. 22 The influence of the approach angle on Bennu visual hull when seasonal effects are present

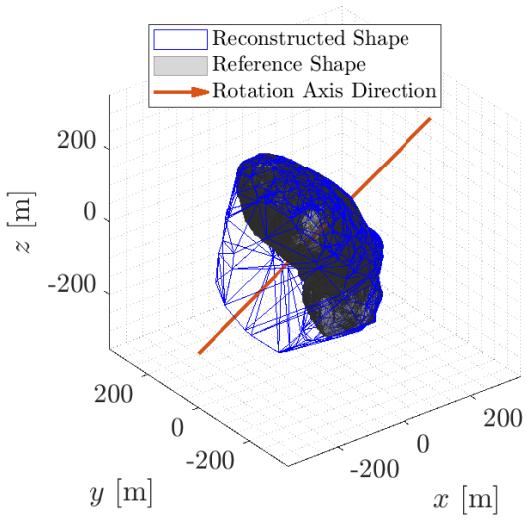


Fig. 23 The influence of the illumination angle on Itokawa visual hull when seasonal effects are present

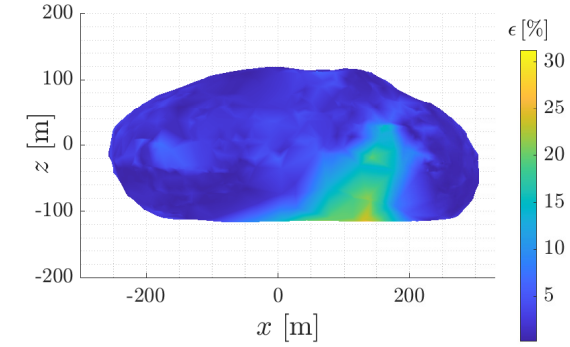
effects. The visual hull reconstruction is shown in Figure 23. In this approach configuration, the north pole is always at light which implies that it is well reconstructed. On the contrary, the south pole is not reconstructed, as not illuminated, and the algorithm compensates for this by intersecting the viewing cones behind the terminator line.

E. Itokawa test case with Hayabusa mission data

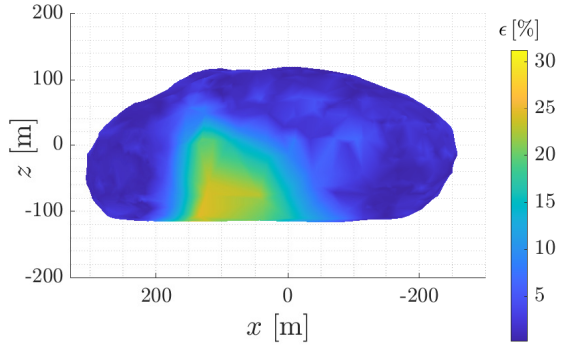
In this subsection, the algorithm is tested on Hayabusa mission data. The observational and illumination geometries are retrieved from mission SPICE kernels. Moreover, images are chosen among the ones taken with the camera green filter during the Hayabusa approach to Itokawa *. The aim of this study is twofold. First, spacecraft poses are affected by orbit determination errors. This affects both the shadow-robust silhouette computation and the shape reconstruction. During the silhouette computation, the tangent Sun rays projections depend on the spacecraft pose, leading to errors in the shadow-robust silhouette extraction. These silhouettes are then intersected by computing the associated viewing cone. The viewing con intersection depends on the spacecraft position and orientation. When the cone intersection is affected by errors, the reconstructed shape changes accordingly. It is worth underlining that the viewing cone intersection is exclusionary. This means that a viewing cone part not intersecting all the other cones does not belong to the reconstructed shape.

Second, images are affected by real camera effects, such as real PSF, detector errors, and optical distortions. An important effect is that small body limbs are spread over several pixels. Thus, Otsu's method detects with less precision the limbs location. Errors in limb location imply the loss of tangency between the viewing cones and the real small

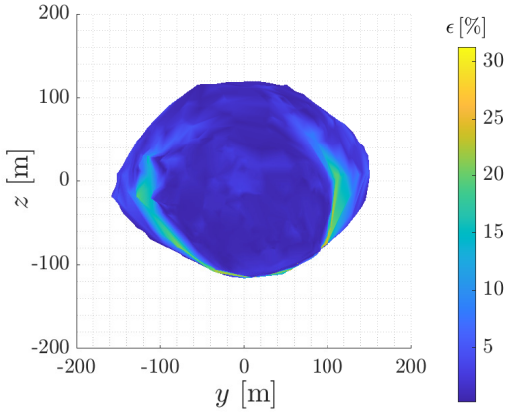
*Chosen images are available at `https://sbnarchive.psi.edu/pds3/hayabusa/HAY_A_AMICA_3_HAYAMICA_V1_0/data/20050930` and `https://sbnarchive.psi.edu/pds3/hayabusa/HAY_A_AMICA_3_HAYAMICA_V1_0/data/20050930`



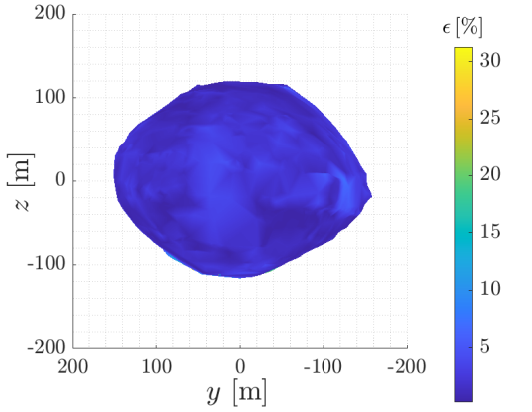
(a) y axis upward



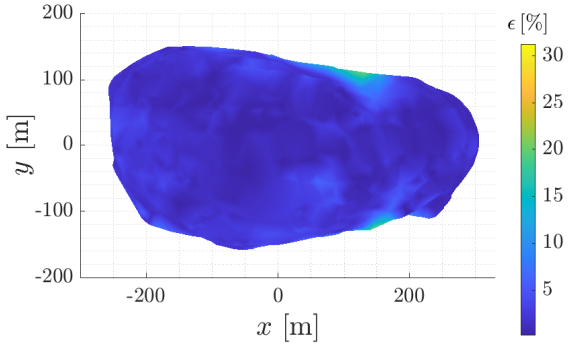
(b) y axis downward



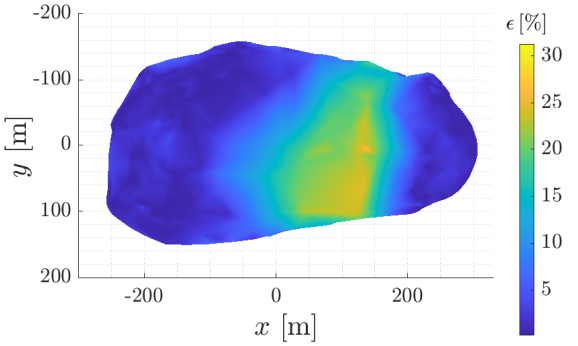
(c) x axis upward



(d) x axis downward



(e) z axis upward



(f) z axis downward

Fig. 24 The reconstruction error ϵ projected on the different Cartesian planes for Hayabusa mission data.

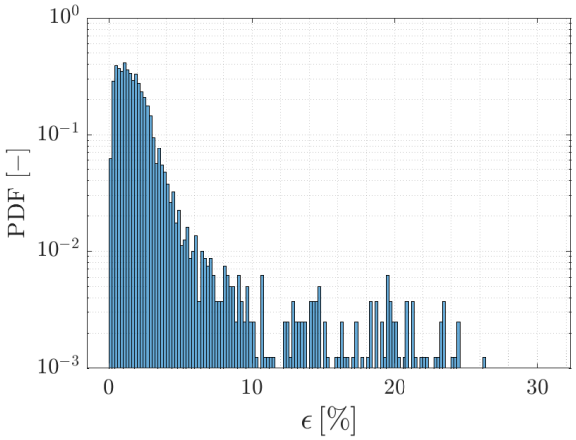


Fig. 25 The histogram of the reconstruction error ϵ for Hayabusa mission data.

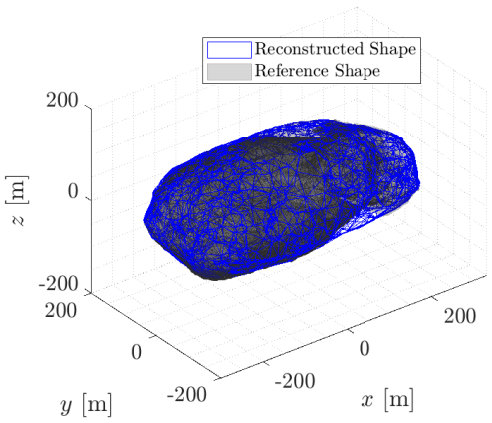


Fig. 26 Reconstructed visual hull for Hayabusa mission data.

body shapes, leading to visual hull reconstruction errors.

Despite the presence of these additional errors, the algorithm reconstructs the asteroid shape with performances similar to the synthetic test case reported in Section VII.C. The detailed map of the error is shown in Figures 24a - 24f on the Itokawa reconstructed shape, while Figure 25 shows the normalized Hausdorff distance PDF. Note that the color bar limits are consistent with Figures 19a - 19f. The reconstruction obtained from the algorithm is reported in Figure 25. As in the synthetic case, the reconstruction, shown in Figure 26, well agrees on the two lobes as they are directly illuminated by the Sun during the observational period. The north and south pole concavities are not reconstructed, as in Section VII.C, as they are not directly illuminated. Moreover, it is worth underlining that the reconstruction error is lower than the synthetic case at the pole as the spacecraft is not constrained to an in-plane trajectory. Indeed, the spacecraft movement out of the asteroid rotation plane implies that the poles are more constrained, thus better reconstructed. For this reason, the relative volumetric error is 15.21% which is lower than the synthetic case (see Table 3). Analogously, the maximal error, which is 26.35% or $\sim$ 41.8 m, is lower because of the better observability at the poles from the Hayabusa trajectory. The reconstruction statistics are reported in Table 3. The reconstruction error mean and variance are slightly higher than the synthetic test case because the reconstruction is slightly less accurate. This is due to the errors in spacecraft pose and in limb localization which leads to a less accurate visual hull reconstruction. This is also visible from the differences in error distribution between Figure 20 and Figure 25. Note that the limb localization error could be mitigated by implementing a sub-pixel accuracy limb detection algorithm, such as the ones reviewed in Christian [39].

VIII. Discussion

The simulations underline the performance of the proposed algorithm in terms of reconstruction quality. The shadow-robust silhouette extraction is a reliable technique to reconstruct shadows in the image plane. The proposed

shadow-robust silhouette extraction was tested extensively (even with very high illumination angles) by providing satisfactory performance. This enables robust information extraction when the scene is partially illuminated and the small body shape is not known. The 3D points computation algorithm processes the computed shadow-robust silhouettes to reconstruct the small body visual hull. The analysis has shown the applicability of the proposed algorithm to small body approach. Small body shape is well reconstructed with limited errors with favorable approach geometry. If seasonal effects are present, the reconstruction algorithm compensates for the non-observability with a conservative reconstruction.

An important step to reduce the computation time is the heuristic to bound the silhouette points number. In the previous simulations, the initial vertices are more than one thousand, leading to a high computational burden . By reducing the silhouette points to 100, the computational time decreases with limited errors in the reconstruction. Note that this is not the number of points on the shadow-robust silhouette, but the maximal number of points in each initial polygon. The number of points of the shadow-robust silhouette is variable over the different silhouettes as some points are removed. It is worth analyzing the evolution of the computational time by increasing the average points number in the shadow-robust silhouettes. Figure 27 shows the obtained numerical results and the associated linear and quadratic fittings. This is

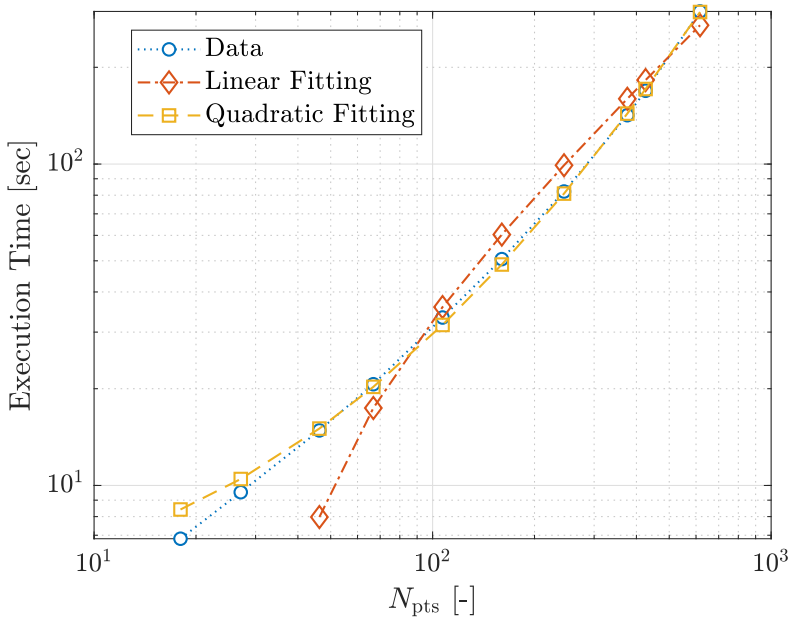


Fig. 27 The computational time by varying the average number of points in the polygon.

in accord with computational complexity as studied in Boyer and Franco [22]. Indeed, the complexity of the shape reconstruction is $O\left(N_{\text{im}}^2 N_{\text{pts}} r\right)$ where N_{im} is the image number, N_{pts}^2 the average vertices number in the silhouettes, and r is the plane-ray intersection complexity. The plane-ray intersection complexity is $O\left(N_{\text{pts}}\right)$ in the worst case and $O(1)$ in the best implementation [22]. As the Ball-Pivoting algorithm complexity is linear with respect to the input 3D points

[31], the overall algorithm worst-case complexity is $O\left(N_{\text{im}}^2 N_{\text{pts}}^2\right)$. Note that the shadow-robust silhouette computation has not been considered because less time-consuming than the visual hull reconstruction. As the number of images N_{im} is generally increased to obtain a better approximation of the visual hull, the number of points N_{pts} is a key parameter to reduce the computational burden of the proposed algorithm. Note that, as underlined by Boyer and Franco [22], an optimal combination of images and points exists that provide accurate results. A numerical analysis, not reported here for the sake of brevity, has shown that the error histograms in Figures 17 and 20 are barely affected by the points reduction as the highest errors are due to the difference between the visual hull and the true shape.

Moreover, it is worth noting that the current edge detection technique, i.e., the binarization with Otsu's method, has pixel accuracy in determining the edge location. Future work should investigate the improvements in reconstruction accuracy when using sub-pixel edge detection techniques. The correct edge determination implies a reduction of silhouette determination error and, consequently, of the shape reconstruction error.

Note that it has been assumed in the paper that the optics is modeled by a distortion-free pin-hole projection with known calibration. A more detailed study should be performed to simulate a more detailed sensor, include errors in the calibration, and understand the sensitivity to the PSF. This leads to understanding the algorithm limits and its applicability in real applications. The algorithm is expected not to be extremely sensitive to variation in albedo maps or BRDF as the silhouette extraction uses images binarization to identify the initial polygons.

Finally, the algorithm is not suited to have multiple objects in the field of view as in asteroid binary systems. Future work should investigate this scenario to improve the algorithm applicability. This issue is resolved by storing the plane portions that do not belong to the silhouettes. Additional details are available in Franco and Boyer [20].

IX. Conclusion

This paper outlines the design of a shape from silhouette algorithm for small-body exploration. The algorithm is composed of two different parts: the shadow-robust silhouette extraction and the 3D reconstruction from the silhouettes. The shadow-robust silhouette extraction takes into account the geometry between the Sun, the small body, and the camera to reconstruct the shadow projections and isolate the illuminated limbs. This ensures that all the exploited information belongs to the small body limbs and avoids problems due to self-shadowing and terminator-line shadows. Once the shadow-robust silhouettes are extracted, their segments are backprojected in the 3D space to find the small-body visual hull as the intersection of the different viewing cones.

Numerical simulations with convex and concave shapes and under different illumination conditions have been presented with favorable obliquity. The simulations underline that a high-precision visual hull reconstruction is computed for a convex body despite a small image dataset and adverse illumination conditions. In the case of a concave shape, the visual hull reconstruction is thorny since not all concavities can be inferred. The computed shape is an excellent first guess for other types of algorithms, like shape from shading, or a good low-resolution approximation for model-based

tracking algorithms. Then the reconstruction solutions with adverse obliquity are shown to understand the algorithm behavior in this challenging situation. Finally, the algorithm is tested on real mission data with results comparable with the synthetic ones.

The proposed techniques can be used to determine the visual hull and constrain the asteroid shape during the approach.

A. Intersection between a plane and a ray

The intersection of a ray and a plane is computed thanks to standard techniques of the ray-tracing community [40]. Let a ray be defined by a 3D point $\boldsymbol{p}_{\text{ray}}$ and a direction $\hat{\boldsymbol{v}}_{\text{ray}}$. Let a plane be defined by a 3D point $\boldsymbol{p}_{\text{pl}}$ and the plane normal $\hat{\boldsymbol{n}}_{\text{pl}}$. The 3D intersection point between the ray and the plane $\boldsymbol{p}_{\text{int}}$ is given by:

$$\boldsymbol{p}_{\text{int}} = \boldsymbol{p}_{\text{ray}} + \hat{\boldsymbol{v}}_{\text{ray}} t \qquad \text{where} \qquad t = \frac{\hat{\boldsymbol{n}}_{\text{pl}}^T \left(\boldsymbol{p}_{\text{pl}} - \boldsymbol{p}_{\text{ray}} \right)}{\hat{\boldsymbol{n}}_{\text{pl}}^T \hat{\boldsymbol{v}}_{\text{ray}}} \tag{3}$$

Acknowledgment

This research received funding from the CNES, the ISAE-SUPAERO and Airbus Defence & Space under the doctoral contract CNES-2879.

References

[1] Petit, A., *Robust visual detection and tracking of complex objects: applications to space autonomous rendez-vous and proximity operations (PhD Thesis)*, Université Rennes 1, Rennes, France, 2013.

[2] Bercovici, B., and McMahon, J. W., "Robust Autonomous Small-Body Shape Reconstruction and Relative Navigation Using Range Images," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 7, 2019, pp. 1–16. https://doi.org/10.2514/1.G003898.

[3] Feldhacker, J. D., Syal, M. B., Jones, B. A., Doostan, A., McMahon, J., and Scheeres, D. J., "Shape dependence of the kinetic deflection of asteroids," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 10, 2017, pp. 2417–2431. https://doi.org/10.2514/1.G002270.

[4] Scheeres, D. J., Williams, B. G., Bollman, W. E., Davis, R. P., Helfrich, C. E., Synnott, S. P., and Yeomans, D. K., "Navigation for low-cost missions to small solar-system bodies," *Acta Astronautica*, Vol. 35, 1995, pp. 211–220. https://doi.org/10.1016/0094-5765(94)00186-P.

[5] Werner, R. A., and Scheeres, D. J., "Exterior gravitation of a polyhedron derived and compared with harmonic and mascon gravitation representations of asteroid 4769 Castalia," *Celestial Mechanics and Dynamical Astronomy*, Vol. 65, No. 3, 1996, pp. 313–344. https://doi.org/10.1007/BF00053511.

[6] Panicucci, P., Bercovici, B., Zenou, E., McMahon, J., Delpech, M., Lebreton, J., and Kanani, K., "Uncertainties in the gravity

spherical harmonics coefficients arising from a stochastic polyhedral shape," *Celestial Mechanics and Dynamical Astronomy*, Vol. 132, No. 4, 2020, pp. 1–27. https://doi.org/10.1007/s10569-020-09962-8.

[7] Bercovici, B., Panicucci, P., and McMahon, J., "Analytical shape uncertainties in the polyhedron gravity model," *Celestial Mechanics and Dynamical Astronomy*, Vol. 132, No. 5, 2020, pp. 1–32. https://doi.org/10.1007/s10569-020-09967-3.

[8] Gaskell, R. W., Barnouin-Jha, O. S., Scheeres, D. J., Konopliv, A. S., Mukai, T., Abe, S., Saito, J., Ishiguro, M., Kubota, T., Hashimoto, T., Kawaguchi, J., Yoshikawa, K., Kominato, T., Hirata, N., and Demura, H., "Characterizing and navigating small bodies with imaging data," *Meteoritics & Planetary Science*, Vol. 43, No. 6, 2010, pp. 1049–1061. https://doi.org/10.1111/j.1945-5100.2008.tb00692.x.

[9] Capanna, C., Gesquière, G., Jorda, L., Lamy, P., and Vibert, D., "Three-dimensional reconstruction using multiresolution photoclinometry by deformation," *The Visual Computer*, Vol. 29, 2013, pp. 825–835. https://doi.org/10.1007/s00371-013-0821-5.

[10] Gaskell, R. W., "Automated landmark identification for spacecraft navigation," *AAS/AIAA Astrodynamic Specialist Conference*, 2001, pp. 1–11.

[11] Kanani, K., Robin, C., Masson, A., Brochard, R., Duteis, P., and Delage, R., "Dual technologies to mature vision based navigation: the example of space rendezvous and ait-to-air refueling," *10th International ESA conference on Guidance, Navigation and Control Systems*, 2016, pp. 1–11.

[12] Takahashi, S., and Scheeres, D. J., "Autonomous Navigation and Exploration of a Small Near-Earth Asteroid," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 4, 2021, pp. 701–718. https://doi.org/10.2514/1.G005733.

[13] Hartley, R. I., and Zisserman, A., *Multiple view geometry in computer vision*, Cambridge university press, 2004. https://doi.org/10.1017/CBO9780511811685.

[14] Harris, C., and Stephens, M., "A combined corner and edge detector," *Alvey vision conference*, 1988, pp. 1–5.

[15] Panicucci, P., *Autonomous vision-based navigation and shape reconstruction of an unknown asteroid during approach phase (PhD Thesis)*, ISAE-SUPAERO, Toulouse, France, 2021.

[16] Dor, M., Skinner, K. A., Driver, T., and Tsiotras, P., "Visual SLAM for Asteroid Relative Navigation," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, 2021, pp. 2066–2075. https://doi.org/10.1109/CVPRW53098.2021.00235.

[17] Savarese, S., Andreetto, M., Rushmeier, H., Bernardini, F., and Perona, P., "3D reconstruction by shadow carving: Theory and practical evaluation," *International journal of computer vision*, Vol. 71, No. 3, 2007, pp. 305–336. https://doi.org/10.1007/s11263-006-8323-9.

[18] De Santayana, R. P., and Lauer, M., "Optical measurements for rosetta navigation near the comet," *25th International Symposium on Space Flight Dynamics*, 2015, pp. 1–19.

[19] Bandyonadhyay, S., Nesnas, I., Bhaskaran, S., Hockman, B., and Morrell, B., "Silhouette-Based 3D Shape Reconstruction of a Small Body from a Spacecraft," *IEEE Aerospace Conference*, IEEE, 2019, pp. 1–13. https://doi.org/10.1109/AERO.2019.8741753.

[20] Franco, J.-S., and Boyer, E., "Efficient polyhedral modeling from silhouettes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31, No. 3, 2008, pp. 414–427. https://doi.org/10.1109/TPAMI.2008.104.

[21] Franco, J.-S., and Boyer, E., "Exact polyhedral visual hulls," *British Machine Vision Conference*, BMVA Press, 2003, pp. 32.1–32.10. https://doi.org/10.5244/C.17.32.

[22] Boyer, E., and Franco, J.-S., "A hybrid approach for computing visual hulls of complex objects," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 2003, pp. 695–701. https://doi.org/10.1109/CVPR.2003.1211421.

[23] Baker, D. A., and McMahon, J. W., "Limb-Based Shape Modeling: A Demonstration on Itokawa," *2nd RPI Space Imaging Workshop*, 2019, pp. 1–10.

[24] Panicucci, P., McMahon, J., Zenou, E., Delpech, M., Lebreton, J., and Kanani, K., "Polyhedral Shape From Silhouettes for Small Bobdy Chacterization," *43rd Rocky Mountain AAS Guidance, Navigation and Control Conference*, 2020, pp. 1–12.

[25] Laurentini, A., "The visual hull concept for silhouette-based image understanding," *IEEE Transactions on pattern analysis and machine intelligence*, Vol. 16, No. 2, 1994, pp. 150–162. https://doi.org/10.1109/34.273735.

[26] Laurentini, A., "How far 3D shapes can be understood from 2D silhouettes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 2, 1995, pp. 188–195. https://doi.org/10.1109/34.368170.

[27] Panicucci, P., Brochard, R., Lebreton, J., Lefez, R., Zenou, E., and Delpech, M., "Localization And Mapping Merging Silhouttes Information And Feature Tracking For Small Body Applications," *11th International ESA conference on Guidance, Navigation and Control Systems*, 2021, pp. 1–15.

[28] Otsu, N., "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, Vol. 9, No. 1, 1979, pp. 62–66. https://doi.org/10.1109/TSMC.1979.4310076.

[29] Christian, J. A., "A tutorial on horizon-based optical navigation and attitude determination with space imaging systems," *IEEE Access*, Vol. 9, 2021, pp. 19819–19853.

[30] Matusik, W., Buehler, C., Raskar, R., Gortler, S. J., and McMillan, L., "Image-based visual hulls," *27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 369–374. https://doi.org/10.1145/344779.344951.

[31] Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., and Taubin, G., "The ball-pivoting algorithm for surface reconstruction," *IEEE transactions on visualization and computer graphics*, Vol. 5, No. 4, 1999, pp. 349–359. https://doi.org/10.1109/2945.817351.

[32] Ishiguro, M., Nakamura, R., Tholen, D. J., Hirata, N., Demura, H., Nemoto, E., Nakamura, A. M., Higuchi, Y., Sogame, A., Yamamoto, A., Kitazato, K., Yokota, Y., Kubota, T., Hashimoto, T., and Saito, J., "The Hayabusa spacecraft asteroid multi-band imaging camera (AMICA)," *Icarus*, Vol. 207, No. 2, 2010, pp. 714–731. https://doi.org/10.1016/j.icarus.2009.12.035.

[33] Rizk, B., Drouet d'Aubigny, C. Y., Golish, D., Fellows, C., Merrill, C., Smith, P., Walker, M. S., Hendershot, J. E., Hancock, J., Bailey, S. H., DellaGiustina, D. N., Lauretta, D. S., Tanner, R., Williams, M., Harshman, K., Fitwgibbon, M., Vertw, W., Chen, J., Connors, T., Hamara, D., Dowd, A., Lowman, A., Dubin, M., Burt, R., Whiteley, M., Watson, M., McMahon, T., Ward, M., Booher, D., Read, M., Williams, B., Hunten, M., Little, E., Saltzman, T., Alfred, D., O'Dougherty, S., Walthall, M., Kenagy, K., Peterson, S., Crowther, B., Morgan, M. L., Castle, C., Dominguez, R., and Sullivan, M., "OCAMS: the OSIRIS-REx camera suite," *Space Science Reviews*, Vol. 214, No. 1, 2018, p. 26. https://doi.org/10.1007/s11214-017-0460-7.

[34] Lebreton, J., Brochard, R., Baudry, M., Jonniaux, G., Salah, A. H., Kanani, K., Le Goff, M. L., Masson, A., Ollagnier, N., Panicucci, P., Proag, A., and Robin, C., "Image simulation for space applications with the SurRender software," *arXiv preprint arXiv:2106.11322*, 2021.

[35] Hapke, B., *Theory of reflectance and emittance spectroscopy*, Cambridge university press, 2012. https://doi.org/10.1017/CBO9781139025683.

[36] 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., Miyamato, H., and Hirata, N., "Pole and global shape of 25143 Itokawa," *Science*, Vol. 312, No. 5778, 2006, pp. 1347–1349. https://doi.org/10.1126/science.1126574.

[37] Barnouin, O. S., Daly, M. G., Palmer, E. E., Gaskell, R. W., Weirich, J. R., Johnson, C. L., Al Asad, M. M., Roberts, J. H., Perry, M. E., Susorney, H. C. M., Daly, R. T., Bierhaus, E. B., Seabrook, J. A., Espiritu, R. C., Nair, A. H., Nguyen, L., Neumann, G. A., Ernst, C. M., Boynton, W. V., Nolan, M. C., Adam, C. D., Moreau, M. C., Rizk, B., Drouet d'Aubigny, C. Y., Jawin, E. R., Walsh, K. J., Michel, P., Schwartw, S. R., Ballouz, R.-L., Mazarico, E. M., Scheeres, D. J., McMahon, J. W., Bottke, W. F., Sugita, S., Hirata, N., Hirata, N., Watanabe, S.-i., Burke, K. N., DellaGiustina, D. N., Bennett, C. A., Lauretta, D. S., and The OSIRIS-REx Team, "Shape of (101955) Bennu indicative of a rubble pile with internal stiffness," *Nature Geoscience*, Vol. 12, No. 4, 2019, pp. 247–252. https://doi.org/10.1038/s41561-019-0330-x.

[38] Lauretta, D. S., DellaGiustina, D. N., Bennett, C. A., Golish, D. R., Becker, K. J., Balram-Knutson, S. S., Barnouin, O. S., Becker, T. L., Bottke, W. F., Boynton, W. V., Campins, H., Clark, B. E., Connolly, H. C. J., Drouet d'Aubigny, C. Y., Dworkin, J. P., Emery, J. P., Enos, H. L., Hamilton, V. E., Hergenrother, C. W., Howell, E. S., Izawa, M. R. M., Kaplan, H. H., Nolan, M. C., Rizk, B., Roper, H. L., Scheeres, D. J., Smith, P. H., Walsh, K. J., Wolner, C. W. V., and The OSIRIS-REx Team, "The unexpected surface of asteroid (101955) Bennu," *Nature*, Vol. 568, No. 7750, 2019, pp. 55–60. https://doi.org/10.1038/s41586-019-1033-6.

[39] Christian, J. A., "Accurate planetary limb localization for image-based spacecraft navigation," *Journal of Spacecraft and Rockets*, Vol. 54, No. 3, 2017, pp. 708–730. https://doi.org/10.2514/1.A33692.

[40] Akenine-Moller, T., Haines, E., and Hoffman, N., *Real-time rendering*, AK Peters/crc Press, 2019.