UNIVERSITY OF COPENHAGEN
FACULTY OF SCIENCE

PhD Thesis

3D Reconstruction with Refraction

Robin Bruneau

Supervisors: François Lauze & Jean-Denis Durou
Co-Supervisors: Yvain Quéau & Kim Steenstrup Pedersen

Submitted: 31st July, 2024

This thesis has been submitted to the PhD School of The Faculty of Science, University of Copenhagen

Abstrakt

Rav er forstenet harpiks fra gamle nåletræer. Det kan findes mange steder, og hvis man er forsigtig, kan man finde levn fra fortiden indlejret i det. Uanset om det er planter eller insekter, er disse fossiler millioner af år gamle. I en tid, hvor man studerer udviklingen af den levende verden omkring os, vil entomologer gerne kunne studere disse insekter i detaljer, så de kan forbindes med nutidens insekter. Mens en grundlæggende metode ville være at observere dem under et mikroskop, ville en anden være at skabe en 3D-model af disse insekter, så de lettere kan observeres, sammen med muligheden for at foretage målinger af dem. Denne afhandling undersøger metoder til 3D-rekonstruktion af objekter, der er nedsænket i brydende medier, med den idé, at det komplekse tilfælde med rav vil blive behandlet i det fremtidige arbejde. Forskningen integrerer geometriske og fotometriske metoder samt neurale inverse gengivelsesteknikker for at udvikle robuste og præcise 3D-rekonstruktionspipelines. Det første fokus er en metode baseret på multi-view, som giver et trekantet net af et polyeder ved hjælp af hjørnedetektion og estimering af ellipsoide baner. Med viden om mediets form blev der derefter arbejdet på at tilpasse multi-view stereo (MVS) metoder til refraktion. Samtidig blev tilpasningen af fotometriske stereometoder i tilfælde af en brydende plan grænseflade mellem kameraet og objektet også undersøgt. Endelig foreslog vi efter arbejdet med de neurale metoder til 3D-rekonstruktion en pipeline til 3D-rekonstruktion med høje detaljer ved hjælp af multi-view multi-illuminationsbilleder. Det seneste arbejde, som endnu ikke er helt afsluttet, har fokuseret på at opnå brydningsnormalkort og på multiview-integration af disse normalkort inden for rammerne af en polyedrisk brydningsgrænseflade.

Abstract

Amber is a fossilised resin from ancient conifers. It can be found in many places, and if you are careful, you can discover relics of the past embedded inside. Whether plants or insects, these fossils are millions of years old. At a time when the evolution of the living world around us is being studied, entomologists would like to be able to study these insects in detail so that they can be linked to the insects of today. While one basic method would be to observe them under a microscope, another would be to provide a 3D model of these insects so that they can be observed more easily, along with the ability to take measurements of them. This thesis investigates methods for the 3D reconstruction of objects immersed in refractive media, with the idea that in future work, the complex case of amber will be addressed. The research integrates geometric and photometric methods, as well as neural inverse rendering techniques to develop robust and precise 3D reconstruction pipelines. The first focus is a method based on multi-view, which provides a triangular mesh of a polyhedron using corner detection and ellipsoid trajectory estimation. With the knowledge of the shape of the medium, work was then carried out on adapting multi-view stereo to refraction. At the same time, the adaptation of photometric stereo in the case of a refractive planar interface between the camera and the object was also studied. Finally, following the work on the neural methods for 3D reconstruction, we proposed a pipeline for 3D reconstruction with high details using multi-view multi-illumination images. The latest work, which is not yet fully conclusive, has focused on obtaining refractive normal maps and on the multi-view integration of these normal maps within the framework of a polyhedral refractive interface.

Acknowledgments

First of all, I would like to thank the assessment committee who agreed to read my work and discuss it during my thesis defence.

I would also like to express my gratitude to all the teachers at ENSEEIHT, who enabled me to acquire the knowledge I needed to carry out my research. I would particularly like to thank Professors Maxime Descoteaux and Manuel Lafond from the University of Sherbrooke, who offered me their vision of research and shared their work with me during my school exchange.

The members of the Image team at DIKU and the REVA team at IRIT also deserve thanks for their warm welcome. The time spent discussing research, exchanging views on various subjects, sharing breakfasts, climbing, and having conversations at Le Poêle de la Bête contributed greatly to a favourable atmosphere.

I would also like to thank my friends at ENSEEIHT for the times we shared that helped me relax and think about other research topics. A special thanks to Jéremie and Yann, my former flatmates, for their hospitality during my stays in Toulouse.

My high school friends also brought a valuable outside perspective to my research. Having to explain my work in simple terms has enriched my understanding of it on different scales.

I would of course like to thank my close family for their constant support despite the distance, and for their efforts to understand my work, even though the subject was completely foreign to them. Their reassurance over these three years has been indispensable.

I will end by thanking the people who contributed most to the success of this thesis:

We will start with Lilian Calvet, with whom I have worked on various projects

over these three years, but whom I got to know mainly during our CVPR rush. He is a simple person, always in a good mood and always looking to do things right. This ability to listen to others and guide you through the maze of his labyrinthine ideas makes him a pleasure to work with.

Jean Mélou, who, during his courses at ENSSEIHT, was able to make me understand the importance and benefits of research, both from a personal and professional point of view. A person who is always there to help you, to exchange ideas and also to share good times.

Baptiste Brument, another PhD student in the REVA team, who helped make this adventure even simpler by working together on our projects and sharing our knowledge about new neural methods. This exchange and discussion helped me to familiarise myself with certain principles. I will also remember his enthusiasm for life and the quality time we spent in Bologna and Seattle.

I would also like to thank Yvain Quéau, for the guidance he gave me in my thesis, for his generosity and his active participation in all my contributions. He is a very talented person, who makes excellent suggestions, who knows how to listen to you, but also how to say straightforward things to move in the right direction. I would also like to thank him for the time he took to proofread this thesis, the feedback helped me to propose a much more structured and better-written manuscript.

Jean-Denis Durou, my thesis supervisor at IRIT, is a person who is almost impossible not to like. I would like to thank him for pushing me in the field of research, and for offering to collaborate with him on this project in partnership with Denmark. Our discussions about research, and his belief in a simple, sharing life, put me in the best possible position to complete my thesis. If you add to that his nitpick approach to writing research papers, you get a winning combo of effective research and polished articles.

Finally, I would like to thank François Lauze, my thesis supervisor at DIKU, without whom nothing would have been possible. He was able to help me through our daily exchanges in the overall organisation of this thesis, whether it was to motivate me, to exchange ideas on specific subjects, for administrative problems, for writing or mathematics. But also in everyday life, doing his utmost to make me feel at ease in Denmark. With François, I never had the impression that I was doing my thesis on my own; every phase of this thesis was discussed and was a joint effort. I started these three years with a stranger who had shown me insects in amber at a thesis party in 2020 and ended up with a thesis director who was as understanding and helpful as a member of my family, with whom I was able to show that this dream of reconstructing fossils in amber was not impossible.

Contents

List of Figures

Chapter 1

Introduction

1.1 Context

For those lucky enough to travel to Jutland in northern Denmark, a visit to the picturesque beaches of the region is often a highlight. Among the grains of sand, it is possible to stumble across what is known as the "tears of the gods", more commonly known as amber. These small fossilised resins, with their warm golden hue, gave the colour amber its eponymous name.

Originating from conifers, the creation of amber began millions of years ago. Everything takes shape in this sticky, viscous resin that slowly flows down trunks and branches, sometimes capturing debris, plant fragments, and insects. Over time, the resin hardens and becomes buried under layers of sediment. There, it undergoes chemical transformations. Over the ages, the resin is gradually transformed into amber through processes of polymerisation and fossilisation, ending up as small pieces as we are familiar with today.

What interests us here are the few pieces of amber that contain bits of the past. The plants and insects trapped in the resin are rare remnants of prehistoric species. Entomologists, the scientists who devote their time to the study of insects, seek not only to understand how insects evolve in today's world but also to understand how we arrived at the current diversity of species. This branch of study, devoted specifically to the understanding of living things and how they have evolved, is called phylogeny.

The main aim of phylogeny is to place each living organism in a phylogenetic tree (see Figure 1.1). In the same way as a family tree, where we try to find relatives by going further and further back, a phylogenetic tree consists of grouping individuals with similar characteristics and arranging them in an evolutionary order. If we take a look at the human being "homo sapiens", we

can see that we have an undetermined common ancestor from which the genre homo derives, such as “homo erectus” or “homo habilis”. And if we go even further up the tree, we will find nodes linking us to the pan genre including the bonobo and the chimpanzee. All belonging to the sub-family homininae.

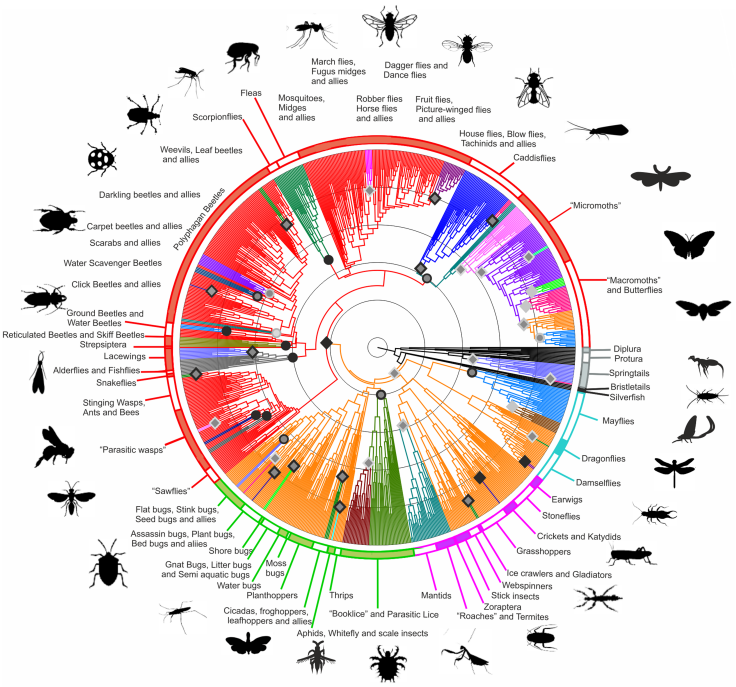


Figure 1.1: Hexapodia phylogenic tree (illustration from [104]).

To be able to reconstruct these trees containing species going back to the beginning of time, it would be necessary in the absolute to be able to have specimens still existing, to be able to study them. Since this hypothesis is impossible, scientists have fallen back on the remains of the past: fossils such as bone remains found in soils, plants or shells moulded in sediments and living organisms trapped in amber.

These pieces of amber contain species ranging from a few million years old to over 90 million years old in the case of Baltic amber. There are many unique species in these small fossilised resins. However, the quality of the pieces, the small size of the insects inside, and other factors mean that the study of amber is progressing very slowly. Therefore, using more recent methods may enable us to study these specimens faster.

To place an insect specimen in a phylogenetic tree, entomologists need infor-

mation about it. This material may be morphological characteristics (wings, legs, eyes, sexual organs, colour, etc.) or molecular characteristics, with the analysis of DNA sequences to understand mutations and differences between the genes in the specimens. This second method, which has become much more accessible recently, is far more accurate because it allows probabilistic calculations to be made to propose the most likely phylogenetic tree. This is in contrast to the morphological classification method, which relies on the direct knowledge of researchers and their ability to describe whether what they observe is closer to specimen A or B. However, the DNA sequencing method has its limitations, and the main one is DNA acquisition.

Deoxyribonucleic acid is not "immortal". DNA has what is known as a half-life of 521 years. So, every 521 years, half the genetic information is lost. In the course of one million years, there are 1919 periods of 521 years, so the probability of there being any remaining is $1/(2^{1919})$, i.e. one chance in 10^{578} (approximately), a number far greater than the number of atoms in the universe (10^{80}) and far greater than googol (10^{100}), which is already excessively huge. Indeed, if you listen to Jérôme Cottanceau: *"What is one googol of a second? To find out, take a snail and make it circle the Earth. Each lap should take about four billion seconds. Each time it completes a million revolutions, remove one molecule of water from the Earth. Since there is approximately* 10^{50}*, the Earth will be dry in about* 10^{65} *seconds. At this point, prepare a pancake to celebrate. Then replace each water molecule in its place, and repeat as many times as it takes for the stack of pancakes to reach Proxima Centauri. You will need about* 10^{19} *pancakes, which will take you about* 10^{84} *seconds. Each time a stack is completed, you will play a euro-million grid, before starting all over again. The law of large numbers will ensure that you win approximately once every 140 million grids played. Then continue the process until you have won the jackpot for the 20 millionth time. You can then stop, as a googol of a second will have elapsed".*

Even if this half-life value can vary due to preservation conditions, these extremely large numbers show that it is almost impossible to find even an ounce of DNA on the specimens trapped in amber. Entomologists will have to resort to the first method of morphological analysis. The question might be raised of removing the insect from the amber to observe it, but there are no insects left inside the amber. The external structure of the insect in the process of polymerisation of the amber has also been transformed into a thin layer and the rest of the insect has been reduced to just bones and wings. It is therefore only possible to observe the insect from the outside.

This is where the PHYLORAMA project comes in. This project, funded by the UCPH Data+ project from the University of Copenhagen, is seeking to propose new methods based on 2D and 3D imaging to digitise and analyse

current insects and those present in amber. The work on 3D reconstruction can be adapted to digitally remove amber and provide 3D models of inaccessible insects from 2D images. This is exactly the context for this thesis, with the ideal aim of developing photographic methods for providing 3D models from pieces of real amber.

There are other methods for 3D scanning, such as those using lasers, based on time-of-flight or phase difference, and also methods based on triangulation with the projection of a pattern. However, these methods seemed less suited to microscopic scales.

Another method is to use X-rays, with medical scanners or synchrotrons. The resolution of medical scanners would be sufficient for insects measuring a few centimetres, but their pixel resolution of 100-500 μm is too low for the acquisition of insects measuring millimetres. The synchrotron, with its way lower resolution (1 nm-5 μm), can be used to 3D reconstruct the image of fossils (see Figure 1.2), enabling the amber to be removed without any problem and the fossil inside to be reconstructed with very high accuracy. However, this method cannot capture the various materials constituting the insect and is also relatively expensive. The Natural History Museum in Copenhagen alone contains a collection of 50,000 pieces of amber, so, the time and money required to 3D scan them all disqualify this method as precise as it may be. We had to move in a direction that is affordable and can be easily paralleled if necessary, which prompted a closer examination of photographic 3D reconstruction in our work. These methods only require a photographic camera and, in some cases, the use of lights, which we believe to be the perfect methods.

The primary aim of this thesis, entitled "3D reconstruction with refraction", is to propose methods for reconstructing objects within refractive entities by adapting existing techniques or introducing new approaches. This work primarily focuses on refractive media such as epoxy cuboids, which offer a deep understanding of the complexities associated with refraction without adding other difficulties caused by amber.

1.2 Amber: A complex medium

Amber, as we have seen, is a fossilised resin with an intriguing complexity. This complexity stems naturally from its unique composition and optical properties. Starting a reconstruction directly with amber pieces would be delicate and almost impossible, due to the numerous properties to model. Refraction and reflection, as they seem to be the main properties of the model, were of particular interest to us. In this section, we will see these different properties and understand how they could interfere with 3D reconstruction.

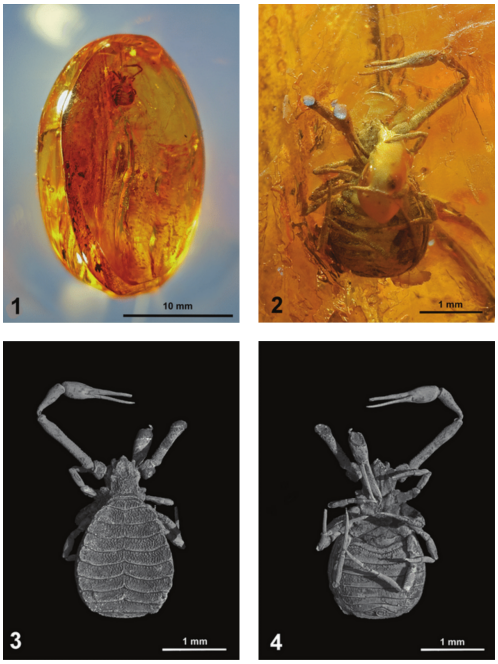


Figure 1.2: Pseudogarypus synchrotron reconstruction. 1) Arachnid trapped in Baltic amber; 2) Magnification of the fossil; 3) 3D reconstruction of the pseudogarypus using the synchrotron at 5.06 μm voxel size (dorsal view); 4) Ventral view. The very high resolution of the synchrotron allows the fine structures and details of the fossil to be recovered while ignoring the amber. Illustration from [44].

1.2.1 Refraction and reflection

Refraction and refractive properties are the cornerstones of our ability to model amber accurately, by explaining the material and its relation in the presence of light. The compression of light rays within amber and the partial reflection of some rays at its surface contribute to the material's transparency and shininess. These properties can be explained by Snell's and Fresnel's laws, which describe how light interacts with the surface of a dielectric material.

Snell's laws: These laws take place in the plane formed with the normal to the surface and the incident ray, which means that the reflected ray, the refracted ray, the incident ray and the surface normal are coplanar (as modelled in Figure 1.3). In this context, the laws relate the angle of incidence i_1 to the angle of refraction i_2 and the angle of reflection i_3 when light interacts from one environment to another:

$$n_1 \sin i_1 = n_2 \sin i_2 \tag{1.1}$$

$$i_1 = -i_3 \tag{1.2}$$

where n_1 and n_2 are the refractive indices of the respective environments.

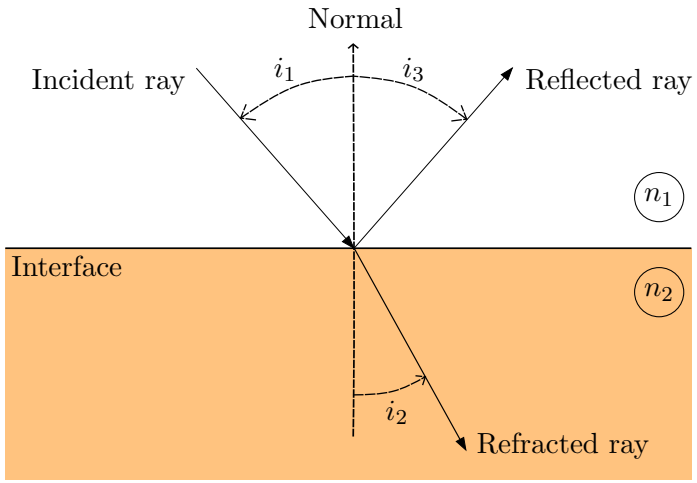


Figure 1.3: Example of the application of Snell's laws with a medium such that $n_2 > n_1$, causing the light ray to bend ($i_2 < i_1$).

In Equation (1.1), n_2 represents the index of refraction (IoR) of the refractive medium. This parameter is strongly related to the light transformation that can be observed. It determines the level of bending of the light direction as it enters or leaves the resin. Amber has a refractive index of around 1.54, which means that light slows down, and its direction bends as it enters the material.

These properties can considerably affect the visual aspect of refractive media. Looking at a specimen of amber found in nature, its arbitrary shape mixing concavities and convexities can cause various visual effects on the insect inside, such as areas that may appear greatly distorted (see Figure 1.4). On the other hand, on pieces that have been reworked by man and cut into a cuboid shape, the insect is seen to be duplicated on all the visible faces. Figure 1.5 shows the impact of different IoRs on the same medium, with not only "order 0" de-multiplications (direct refraction into the visible faces) but also higher order de-multiplications (multiple bounces on the internal faces before reaching the ladybird). Note that the higher the refractive index, the more the rays will bend and, conversely, the more the object will visually tend to move away from the centre of the image. Finally, we can also see that the overall clarity of the ladybird image decreases with the IoR. This is due to the transmittance T coefficient of Fresnel's law, which weakens as the refractive index increases, as can be seen in Figure 1.6.

Figure 1.4: Renderings made with Blender [10] of a graphosoma inside a natural shape with an IoR of 1.54. Large deformations of the insect are noticeable due to refraction.

(a) (b) (c)

Figure 1.5: Renders made with Blender [10] of a ladybird inside a cube with an IoR of 1.33 in (a), of 1.54 in (b) and of 2.0 in (c). The change in IoR can be seen both in the visual distance of the ladybird from the centre of the image and in the loss of clarity.

Fresnel's laws take place in the same context as Snell's laws, however, it quantifies the proportions of light that will refract and reflect at the surface. The transmittance T and reflectance R as they are given by Fresnel's laws, provide coefficients about the intensity evolution when the incident ray hits the refractive surface as follows:

$$R = \frac{1}{2}(R_s + R_p) \tag{1.3}$$

$$T = 1 - R \tag{1.4}$$

$$R_s = \left| \frac{n_1 \cos i_1 - n_2 \cos i_2}{n_1 \cos i_1 + n_2 \cos i_2} \right|^2 \tag{1.5}$$

$$R_p = \left| \frac{n_1 \cos i_2 - n_2 \cos i_1}{n_1 \cos i_1 + n_2 \cos i_2} \right|^2 \tag{1.6}$$

with R_s and R_p corresponding to the reflectance of s-polarised and p-polarised light [5].

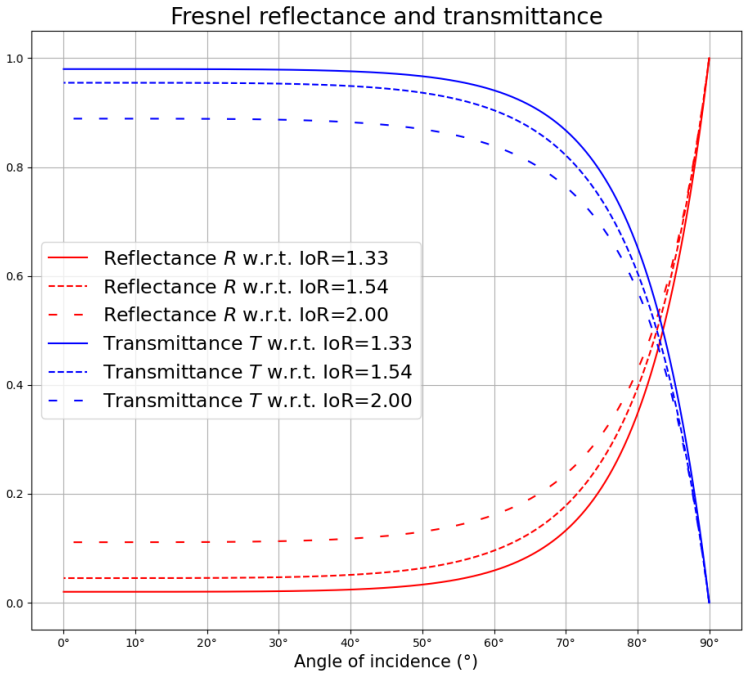


Figure 1.6: Fresnel reflectance R and transmittance T values for different IoRs according to the angle of incidence i_1.

In the case of amber, as can be seen in Figure 1.6, the reflectance at its surface is relatively low, when the incident angle is small, allowing more light to pass through and thus contributing to its transparency. However, when the angle starts to get high ($\geq 60°$), reflectance becomes increasingly important, causing the camera to primarily capture reflections of the environment rather than the interior of the refractive object.

This subsection highlighted the visual and physical effects of refraction and reflection and how they affect images. To show how crucial it is to model these laws, Figure 1.7 illustrates 3D reconstructions of a graphosoma in a cuboid of epoxy with photographic methods ignoring these factors. As expected, they struggle to explain the de-multiplication of the insect, leading to inaccurate geometry.

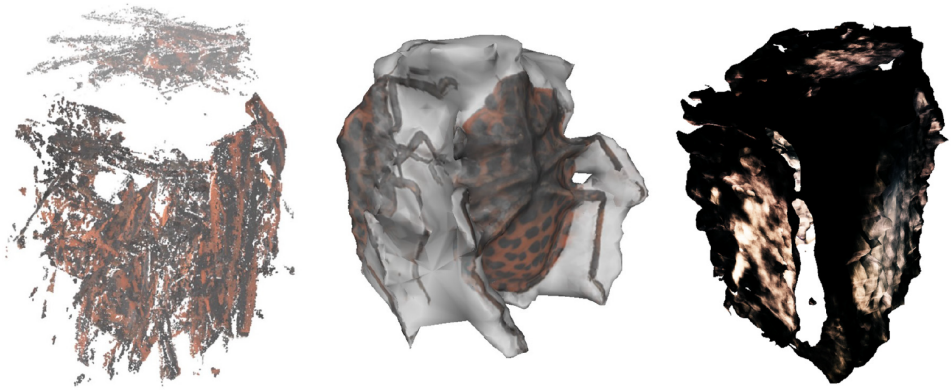


Figure 1.7: Illustrations from [16] representing 3D reconstructions of a graphosoma inside a refractive cuboid without considering refraction. From left to right: A naive multi-view stereo (MVS) method; The MVS approach by Meshroom [38]; The neural 3D surface reconstruction method NeuS2 [123].

Another visual factor that can interfere with 3D reconstruction methods is the colour of amber, far away from the white epoxy model. The next subsection will explain the reason for this colour and the impact on 3D reconstruction.

1.2.2 Composition and colour

When we look at the atomical structure of amber, we can see that it is mainly composed of hydrocarbons, which are organic compounds made up of hydrogen, oxygen and carbon atoms (with chemical formulas varying between $C_{10}H_{16}O$ - $13C_{40}H_{64}O_{14}$ - $12C_{12}H_{20}O$). These different formulas explain the variation in the colour of amber pieces, from pale yellow to dark brown, and sometimes in red, green or blue. Indeed, hydrocarbons explain why amber has a particularly significant absorbance of light at wavelengths around 400-500 nm. These wavelengths correspond to the blue and violet parts of the spectrum, which are more strongly absorbed as explained by Beer-Lambert's law (see Equation (1.7)). And when white light is deprived of its slice of colour around blue, then the warm amber hue remains.

Beer-Lambert's law relates the absorption of light to the properties of the material through which the light travels. It states that absorbance A is directly proportional to the concentration c of the absorbing species and the path length l of the light through the material:

$$A = \epsilon\, c\, l \tag{1.7}$$

where ϵ is the molar absorption coefficient.

Our choice of working with white transparent epoxy pieces allows us to ignore this property due to the absence of colour and also because the absorption of epoxy is in the ultraviolet range. However, for future work where we would like to get closer to a realistic model of amber, we would need to take absorption into account to better explain the colours observed and inverse the absorption process to predict the true colour of the object to be reconstructed. This property is also fundamental in multi-view 3D reconstruction (see Subsection 2.1.5) since the length l travelled in the resin to arrive at a specific 3D point can greatly vary between different views and thus cause colourimetric inconsistency.

This subsection shows that absorption allows us to correctly model the colour of the object to be reconstructed. However, this add-on may not be enough, indeed, when we look at real pieces of amber, the insect is often not the only one that has been trapped in the resin, as we will see in the next subsection dealing with inclusions that can greatly affect the images.

1.2.3 Inclusions

If inclusions in amber carry prehistoric information like insects, these inclusions almost always involve air bubbles as well as dust particles that affect the optical properties of amber. Our methods for 3D reconstruction are based on optical paths (see Section 2.1), so, any small obstruction can break our pipelines. Air bubbles can scatter light, reducing transparency, creating a hazy appearance and altering optical paths. In addition, dust particles on its side, block visibility locally, distorting our perception of the object's colours.

These phenomena will only have a local impact on the quality of the reconstructions. For the time being, we simply neglect them and leave this improvement as perspective.

This last part has shown us a final degree of complexity, allowing us to understand amber and its reaction to visible light. While we can stop at studying the visible spectrum for almost all objects, in the case of amber it is interesting to look at the ultra-violet spectrum too, as we will do in the next subsection.

1.2.4 Fluorescence and UV reaction

When exposed to ultraviolet (UV) light, amber shows a fluorescence property, by emitting visible light. This is induced by the fact that the energy of UV light excites the electrons in the hydrocarbons, moving them to a higher energy level. When these electrons return to their original levels, they release some energy, but this time in the form of visible light (blue or green). Figure 1.8 relates this blue colour emitted by amber when exposed to UV light. The use

of fluorescence is a reliable method for identifying and authenticating amber specimens, and can also be used to obtain low-cost segmentation of the resin in a photographic acquisition framework. When considering that segmentation masks are essential for modelling the shape of amber and applying the laws of refraction and reflection, it is easy to see how this property can have a major impact on the overall 3D reconstruction process.

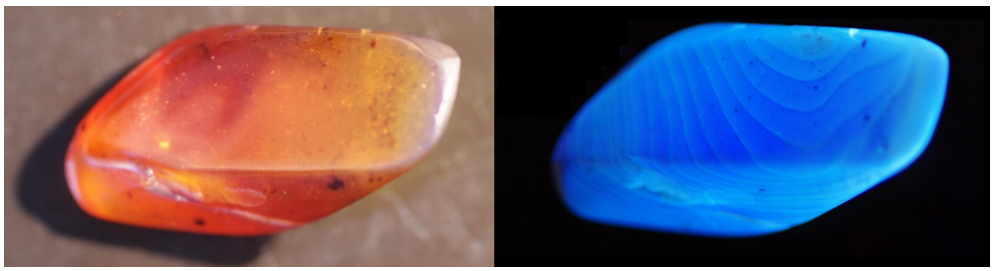


Figure 1.8: Photograph of a piece of amber under natural light (left) and under UV light (right). UV light makes it easier to segment amber based on colour. Pictures from Eric Geirnaert.

1.3 Standing assumptions and thesis structure

In the previous sections, we have looked at the properties of amber, the importance of knowing how to model its shape and being able to use Snell's and Fresnel's laws and model light interactions. We were able to observe how Beer-Lambert's absorption governs its colour, and how simple bubbles and dust can affect our models visually and physically. While our ultimate goal is to be able to digitally remove the amber and reconstruct insects in 3D, this thesis serves first and foremost to show that it is possible to perform this operation despite refraction. We will start by looking at simple refractive shapes (cubes/polyhedrons) with a neutral colour (white), to get rid of absorption and occlusions. In future work, we aim to study more complex models that require applying the absorption law and accounting for bubbles and dust.

Having established the importance of accounting for refraction in 3D reconstruction methods, the remainder of this manuscript will focus on explaining our work, starting with Chapter 2 which will focus on explaining classical photographic 3D reconstruction methods, with Subsection 2.1.5 covering multi-view stereo, Subsection 2.1.6 on photometric stereo, and finally, Section 2.2 on neural methods for 3D reconstruction, highlighting two key methods for our work: NeRF [83] and NeuS [122]. At the end of this chapter, Section 2.3 will discuss current 3D reconstruction methods that incorporate refraction into their models. Next, Chapter 3 will explain our first contribution, a method to determine the 3D geometry of a polyhedron in a multi-view acquisition

setup with a turntable. Chapter 4 will demonstrate how to adapt the classic multi-view stereo method to refraction in the context of a polyhedral or arbitrary medium shape. Chapter 5 will illustrate the adaptation of photometric stereo to an infinite planar refractive interface between two media. Chapter 6 will focus on adapting the NeuS [122] approach to propose a neural method for 3D reconstruction based on multi-view multi-illumination data, enabling high-detail 3D reconstructions. Chapter 7 will explain our current work on photometric stereo in the context of a polyhedral medium, with a focus on obtaining refracted normal maps and integrating these normal maps based on current neural methods. Finally, Chapter 8 will address the issues observed during this thesis, the solutions we propose to address them, and an analysis of future work in this research field.

Chapter 2

Photographic 3D reconstruction

2.1 Classic methods for 3D reconstruction

3D reconstruction is built on several concepts that need to be understood before we can look at conventional 3D reconstruction methods. Knowledge of the different modalities of 3D modelling (see Subsection 2.1.1) will enable us to see their strengths and weaknesses, as well as an understanding of the possible representation of the scene. A broad overview of 3D rendering engines (see Subsection 2.1.2) will give us an insight into how to use the information contained in a scene to render it realistically or not. Then, we will look at all the methods based on photographs for 3D reconstruction (see Subsection 2.1.3), as well as the different camera modalities and how to calibrate them (see Subsection 2.1.4). Finally, we will take a closer look at multi-view stereo (see Subsection 2.1.5) and photometric stereo approaches (see Subsection 2.1.6), which are two pillars of classic 3D reconstruction.

2.1.1 3D modelling

3D modelling is mainly used to study subjects (archaeology, medicine, etc.) or to produce 2D renderings (video games, films, art, etc.). There are many 3D representations, as can be seen in Figure 2.1, with both surface or volumetric representation, in explicit or implicit form.

Surface representations focus on describing the surface of an object. Different modalities can be used to refer to the surface including point clouds, polygonal meshes, parametric surfaces, signed distance functions, etc. Volumetric representations, on the other hand, describe not only the surface but also the interior of an object, using concepts such as voxels or density.

The distinction between implicit and explicit representations is important in 3D modelling. Explicit representations describe surfaces directly, through lists of vertices and polygons, or parametric equations, as is the case for polygonal meshes and parametric surfaces. Implicit representations, on the other hand, use functions to define surfaces and volumes, like the signed distance function which defines the surface of an object by the points where the function vanishes.

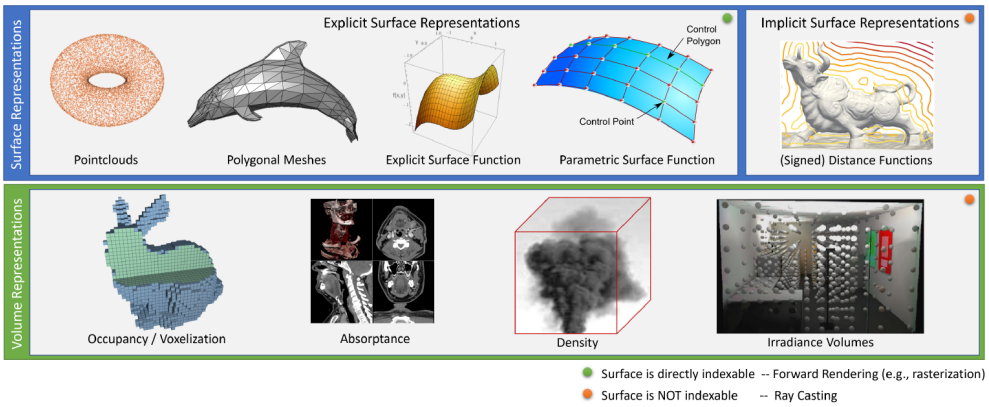


Figure 2.1: An overview of classic surface/volume representations (illustration from [115]).

Here is an in-depth examination of some of the representations used in the contributions of this thesis:

- **Point clouds** consist of a set of discrete points in 3D space, representing the surface of the object without explicit connections between them.
- **Polygon meshes**, on the other hand, connect points by forming polygons, usually triangles or quadrilaterals, thus creating a continuous surface. These meshes are widely used because of their efficiency and simplicity. Outside of the research community, they are a convention for 3D modellers.
- **Signed distance functions** assign to each point in space a value corresponding to its distance from the nearest surface, with a sign indicating whether the point is inside (-) or outside (+) the object. The surface is then defined at the zero level-set of the function.
- The volume representing the scene can be discretised into elementary volumes called **voxels**, which work in the same way as 2D pixels but

extended to a third dimension. They mark space as occupied or free, providing a binary view of occupied space.

- The representation in the form of **density** or **transmittance** refers to the distribution of matter or energy throughout the volume, and is useful for the physical study and simulation of fluids/lights.

2.1.2 3D rendering

In the image-based 3D reconstruction process, cameras are used to generate images of the scene/object. When a camera is used to produce a 2D image, the operation performed is called "photography" or "capture". In computer graphics, the process thing can be done using cameras and a virtual 3D scene, in this case, the operation is called "rendering". The 3D renderers use a "virtual camera" to capture the light emitted and reflected by a 3D scene defined by geometric models and material properties. Current rendering softwares (Blender [10], Maya [4], etc.) allow one to choose between different rendering engines, enabling to juggle between rendering speed and accuracy, depending on the application. Figure 2.2 illustrates the difference between two major techniques, rasterisation and path tracing, the first one being real-time, while the second one allows one to approach impressive realism.

Figure 2.2: Difference of rendering between rasterisation and path tracing (illustration from [97]).

A very popular rendering method for video games is called **rasterisation**, which is a real-time rendering technique. The process involves a polygonal representation of the scene and consists of projecting each triangle of the scene onto the screen plane, and then determining which pixels are covered by that triangle. In addition, lighting and shading models such as Phong [99] or

Blinn-Phong [11] are used to create more realistic renderings. However, rasterisation has its limitations: it cannot handle global illumination effects such as multiple reflections, and refraction is almost impossible to compute due to unknown projection paths induced by refractive interfaces. It is therefore difficult to understand the full complexity of a scene's lighting and to represent the sophisticated materials that are essential for photorealistic rendering.

The second method, called **path tracing**, is an advanced rendering technique that realistically simulates the behaviour of light in a scene by computing the paths that light rays take through it. Unlike rasterisation, which focuses on the rapid projection of geometries onto a 2D plane, path tracing combines simulations of rays leaving the camera, others leaving the light sources and interacting with the surfaces of the scene, to provide results that are as close to reality as possible. This process relies mainly on the probabilistic Monte Carlo model [81], recursivity and Phong bidirectional reflectance model to understand the effects of light scattering, reflection and refraction with a high degree of realism. This method also makes it possible to render complex phenomena such as caustics, self-shadows and global light, offering images of incredible realism.

However, this precision comes at a cost. Path tracing is extremely computationally intensive, making it difficult to use for real-time rendering. Each image requires millions, if not billions, of rays to be traced and their interactions with the surfaces in the scene to be calculated, which takes an extremely long time. Recently, calculation time has been reduced, and a symbiosis between the new GPUs and efficient renderers such as Unreal Engine [121] has made it possible to create video games based on path tracing to run in real time. Although the rendering cost is still very expensive, the combination of numerous optimisations and tricks relating to reflection/refraction made it possible.

2.1.3 Photographic 3D reconstruction methods

Photographic methods for 3D reconstruction can be divided into two groups: techniques based mainly on geometry and those based on photometry. They are mostly found under the generic term shape-from-X, with X representing the approach used to reconstruct a shape from different modalities (silhouette, shading, etc.). As can be seen in Table 2.1, these techniques can be divided into single-view and multi-view methods. It is important to note that while the methods were used separately a decade ago, nowadays geometry and photometry are mixed in multi-view as well as single-view. The new neural approaches that we will see in Section 2.2 no longer carry the name shape-from-X, but many of them are built on the foundations of these classic methods. In [91], it was mentioned that photometric techniques were underestimated, however,

the quality of 3D reconstructions has evolved recently, and the attention to detail and high-frequencies reconstruction offered by these methods has brought them back to the fore. We will see in Subsection 2.1.6 that the information contained in the normals provides one with a considerable deal of value.

	Geometric techniques	Photometric techniques
Single-view techniques ($N = 1$ image)	Projected structured light [35] Shape-from-shadows [108] Shape-from-contour [13] Shape-from-texture [125] Shape-from-template [6]	Shape-from-shading [46]
Multi-view techniques ($N \geq 2$ images)	Structure-from-motion [86] Multi-view stereo [32] Shape-from-silhouette [30] Shape-from-focus [93]	Photometric stereo [100]

Table 2.1: Overview of shape-from-X techniques (taken from [91]).

2.1.4 Camera models

All image-based 3D reconstruction methods require cameras for data acquisition, and also camera calibration to recover 3D modalities from 2D images. To describe the cameras, so-called "extrinsic" and "intrinsic" parameters have been introduced.

2.1.4.1 Extrinsic parameters

The extrinsic parameters explain the position and orientation of the camera in its environment in relation to a reference system that we will refer to as the world reference frame.

We define the world reference frame as an origin Θ_{w} and an orthonormal basis. We define a translation vector $\mathbf{T} \in \mathbb{R}^3$, which describes the position of the origin Θ_{w} of the world system in the camera system.

The orientation of the camera is defined by a rotation matrix $\mathbf{R} \in \mathrm{SO}(3)$, which describes the rotation from the world coordinate system to the camera one. When placed in the camera frame of reference, the x-axis points to the right, the y-axis to the bottom and the z-axis to the front, in the direction in which the camera is looking at, as can be seen in Figure 2.3.

Knowing $\mathbf{R}$ and $\mathbf{T}$ allows us to write the transformation of a 3D point from the world frame to the camera frame and vice versa:

$$X_{\mathrm{c}} = \mathbf{R}\, X_{\mathrm{w}} + \mathbf{T} \tag{2.1}$$

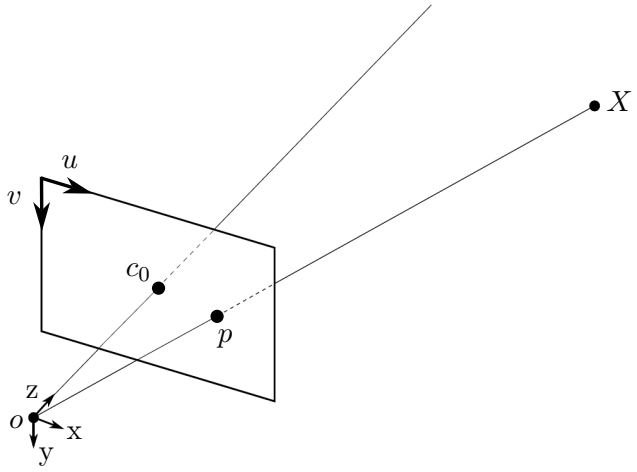


Figure 2.3: Perspective projection of a 3D point X onto a pixel p.

with X_c and X_w representing the 3D point X in the camera and in the world coordinate systems.

2.1.4.2 Intrinsic parameters

The camera's intrinsic parameters explain how the 3D scene seen by the camera is projected into the 2D pixel coordinate system. There are several models of computer vision camera, each with its own parameters.

The most common model is that of the perspective camera, which mimics the behaviour of real cameras. It projects the objects in the scene, taking into account their distance from the camera, which means that distant objects appear smaller than nearby ones. Another model is the orthographic one, where the camera projects objects without taking perspective into account, meaning that objects maintain the same size regardless of their distance from the camera. Such cameras are used in certain cases, even if they are less realistic.

Perspective projection consists of moving from the camera coordinate system to the pixel coordinate system. To do this, we will take the example of a point $X \in \mathbb{R}^3$, whose coordinates are denoted X_w in the world coordinate system and X_c in the camera coordinate system, and compute its projection $p \in \mathbb{R}^2$ expressed in the 2D pixel coordinate system (see Figure 2.3).

The intrinsic parameters are contained in a 3×3 triangular matrix:

$$\mathbf{K} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

which contains five parameters:

- The focal lengths f_x and f_y, expressed in pixel width and pixel height.
- The coordinates c_x and c_y of the principal point c_0 (intersection between the optical axis of the camera and the image plane, see Figure 2.3).
- The skew factor s, which reflects the potential non-orthogonality of the rows and columns of photosensitive electronic cells that make up the camera sensor.

For modern cameras, $f_x = f_y = f$ since the pixels are square, and s is neglected and therefore takes on a zero value. In practice, the calibration matrix $\mathbf{K}$ thus depends only on three parameters:

$$\mathbf{K} = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.3}$$

This allows one to relate, on the one hand, the projection $\tilde{X}_\mathrm{c} \in \mathbb{R}^3$ on the plane of equation $z = 1$ of a 3D point X with coordinates $X_\mathrm{c} = [x_\mathrm{c}, y_\mathrm{c}, z_\mathrm{c}]^\top$ in the camera coordinate system:

$$\tilde{X}_\mathrm{c} = \frac{X_\mathrm{c}}{z_\mathrm{c}} = \begin{bmatrix} \pi(X_\mathrm{c}) \\ 1 \end{bmatrix} \tag{2.4}$$

with the following definition of transformation π:

$$\pi\left([a, b, c]^\top\right) = \left[\frac{a}{c}, \frac{b}{c}\right]^\top \tag{2.5}$$

and, on the other hand, the projection $p = [u, v]^\top \in \mathbb{R}^2$ of X on the image plane, expressed in the pixel coordinate system. If $\tilde{p} = [u, v, 1]^\top \in \mathbb{R}^3$ denotes the homogeneous coordinates of p, this relation is written:

$$\tilde{p} = \mathbf{K}\,\tilde{X}_\mathrm{c} \tag{2.6}$$

Considering perspective projection with extrinsic parameters $\mathbf{R}$ and $\mathbf{T}$, and a calibration matrix $\mathbf{K}$, we call Π the projection operator which allows us to switch from the world coordinates X_w of a 3D point X to the pixel coordinates $p = [u, v]^\top$ of its projection:

$$p = \Pi(X_\mathrm{w}) \tag{2.7}$$

according to the following sequence:

1. We move from the world coordinate system into the camera coordinate system using (2.1);
2. We project X onto the image plane defined at $z = 1$ using (2.4);
3. We then move into the pixel coordinate system using (2.6);
4. In the end, we obtain the coordinates $p = \pi(\tilde{p})$ in the pixel frame using the definition (2.5) of π.

Combining (2.3), (2.4) and (2.5), the last three steps of this sequence give:

$$p = \begin{bmatrix} f\frac{x_c}{z_c} + c_x \\ f\frac{y_c}{z_c} + c_y \end{bmatrix} \tag{2.8}$$

2.1.4.3 Estimation of camera parameters

Intrinsic parameters

A commonly used method for calibrating intrinsic parameters is to insert known patterns in the scene, such as planar grids [119, 138] or spheres [43]. The idea consists of taking photographs of these patterns from different angles and distances. Then, the known reference points of the pattern (intersections between cells, centre/circle of circles) can be related to establish correspondences between the 2D image and the 3D structure. Finally, by applying optimisation algorithms, the values that minimise the projection errors can be found.

While these techniques are still widely used, as they allow an excellent calibration of intrinsic parameters with a large number of views, studies using neural networks to automatically determine these parameters are now also considered [12, 68]. These methods offer more freedom when it comes to data acquisition by making it possible to "model" the different conditions of illumination and possible shadows that could occult certain parts. However, the results are still inferior to traditional methods.

Extrinsic parameters

The estimation of extrinsic parameters, i.e. camera positions and orientations, is generally determined using conventional methods, based either on calibration patterns (checkerboard [138] or ArUco [34]), using the observed deformation of the patterns. With the same idea of detection, but this time with unknown patterns, we can detect common features [107, 8, 74, 135, 28, 7] in different images. Then, these 2D related points allows one to estimate the

fundamental and essential matrices using epipolar geometry [42, 41] and to deduce the extrinsic parameters. Perspective-n-point can also be used, as in [62]. Recent methods based on neural networks [59, 106] manage to approximate the ground truth positions/orientations of cameras, but are still less effective than classic approaches.

All these methods can then be combined with bundle adjustment [118] to optimise the camera settings by using the observed features and their 3D positioning, trying to minimise their projection errors on all the images.

2.1.5 Multi-view 3D reconstruction

Multi-view stereo (MVS) is a classic 3D reconstruction method based on a set of views of a scene, exploiting the epipolar geometry between the overlap areas as well as the colourimetric consistency between the views, in order to propose a 3D model of the scene.

Inputs

Considering a set of N views of the scene, for each view $i \in \{1...N\}$, a rotation matrix $\mathbf{R}_i$, a translation vector $\mathbf{T}_i$, a calibration matrix $\mathbf{K}_i$ and an RGB image I_i are defined. Most of the time, this is supplemented by a binary segmentation mask B_i of the scene to be reconstructed.

Given the first view referred to as the "reference view", we can define the group of views $i \in \{2...N\}$ as the "target views". When no index is used, this means that we refer to the reference view, like I instead of I_{ref}.

Let us take a pixel $p = [u, v]^\top \in B = B_{\text{ref}}$ belonging to the binary mask of the reference view. To obtain the 3D point X which projects in this pixel, with coordinates X_{w} in the world coordinate system, we will use the inverse $\Pi^{-1} = \Pi_{\text{ref}}^{-1}$ of the camera projection defined in (2.7):

$$X_{\text{w}} = \Pi^{-1}(p) \tag{2.9}$$

Equation (2.4) shows that the projection process results in the loss of any notion of depth concerning p. However, X lies on the ray starting from the optical centre o of the camera and passing through the projection of X on the plane of equation $z = 1$, whose coordinates in the camera frame is written, according to (2.6):

$$\tilde{X}_{\text{c}} = \mathbf{K}^{-1}\,\tilde{p} \tag{2.10}$$

We define the following unitary vector supporting this ray:

$$d = \frac{\tilde{X}_c - o}{\left\| \tilde{X}_c - o \right\|} \tag{2.11}$$

so that for each $z \in \mathbb{R}^+$, $X_z = o + zd$ is a plausible candidate for the 3D point X. We thus introduce the following notation:

$$\Pi_z^{-1}(p) = o + zd \tag{2.12}$$

The idea of MVS consists in reprojecting each candidate in all the target views, and to define a score to evaluate the depth. This is where colourimetric consistency comes into play, i.e. the fact that the colour of an object seen through different views should not change significantly.

For each depth z, the candidate 3D point X_z is projected with the Π_i operators of the target views $i \in \{2...N\}$. The colour $I_i \circ \Pi_i \circ \Pi_z^{-1}(p)$ of point X_z in target image i is compared to its colour $I(p)$ in the reference image. This comparison can be made for instance with a norm L1 or L2, giving us a dissimilarity score between the pixels. The sum of these colour discrepancies applied to all target views gives an overall score for the tested depth. The selected 3D point $\hat{X} = o + \hat{z}d$ corresponds to the depth $\hat{z}$ giving the lowest score:

$$\hat{z} = \underset{z \in \mathbb{R}^+}{\operatorname{argmin}} \sum_{i=2}^{N} \left\| I_i \circ \Pi_i \circ \Pi_z^{-1}(p) - I(p) \right\| \tag{2.13}$$

Problem (2.13) can be solved by carrying out an exhaustive search. This "brute force" approach can be reduced using preliminary results such as a shape-from-silhouette [30].

This process is repeated for each pixel of the reference view, and can also be performed on different sets of views, enabling a dense point cloud of the 3D scene to be reconstructed. This explanation is the most basic version of MVS. As we will see in Chapter 4, it is not necessary to go any further to understand our work on refraction. However, most of the time, the point cloud is then transformed into a mesh using Poisson methods [58], or combinations of techniques as proposed by the Meshroom tool by AliceVision [38]. Figure 2.4 shows an example of 3D reconstruction via MVS using Meshroom. We can see that this exhaustive search-based method allows reconstructions of dense and high-quality 3D models. However, it should not be forgotten that this method is not very robust to non-textured scenes. In the case of a uniformly coloured wall, the colour consistency will have great difficulty in finding the right depths.

(a) (b) (c)

Figure 2.4: Reconstruction of the "Wooden lion" from Meshroom [38] using 50 views: (a) Triangle mesh with a high density of triangles due to the high density of points from MVS; (b) Shaded version of the mesh without texture; (c) Textured version of the mesh. The results are of very high quality, with many details visible on the illuminated version of the mesh.

Although very simplistic, this brute-force method is very effective and has been enhanced by work on matching and photometric minimisation [32], on the management of large scenes [116], on the use of depth assumptions [17], on the use of visibility maps [79], and on many other aspects.

2.1.6 Multi-illumination 3D reconstruction

The first work to use shading was carried out by Horn [46] in 1970. This method, known as shape-from-shading, consists of recovering the 3D of an object based on a single view and a single illumination. However, it is very limited, as the single illumination cannot resolve geometric ambiguities. For this reason, the work carried out by Woodham [127] focused on photometric stereo (PS), which is a multi-illumination shape-from-shading method.

Unlike multi-view methods, which offer a global reconstruction of the object with a good understanding of low frequencies, PS only reconstructs a limited part of the object but can produce fine details in the reconstructions. This method is also the only method that allows one to estimate different reflectances, enabling a large spectrum of scenes to be reconstructed.

Input

The most basic case of photometric stereo assumes that the surface $\mathcal{S}$ of the object to be modelled is of Lambertian nature, i.e. the reflection of light is perfectly diffuse. At any point on the surface, an albedo value $\rho \in [0, 1]$ for greyscale images, or $\rho \in [0, 1]^3$ for colour images, can be defined.

Looking at an image I, representing a scene illuminated by a single direc-

tional light $s = [s_x, s_y, s_z]^\top$, we choose a pixel p as the projection of a point $X \in \mathcal{S}$, and we note $n(X)$ the normal to surface $\mathcal{S}$ at X and $\rho(X)$ its associated albedo. The Lambertian assumption gives us the colour $I(p)$ observed in p as a function of $\rho(X)$, $n(X)$ and s:

$$I(p) = \max\{0, \rho(X)\langle n(X), s\rangle\} \tag{2.14}$$

The max operator is used to model shadows, as the surface does not perceive light when $\langle n(X), s\rangle \leq 0$. Since the measurement of the quantity of light cannot be negative, a lower bound must be applied.

Normal and albedo prediction

Assuming $M \geq 3$ illuminations of the scene, $(I_j)_{j\in\{1\dots M\}}$ represent the different images acquired and $(s_j)_{j\in\{1\dots M\}}$ the associated directional lights. In a context of greyscale images, i.e. $\rho \in [0, 1]$, Equation (2.14) for the M different illuminations, results in the following system:

$$\mathcal{P}\begin{cases} I_1(p) = \max\{0, \rho(X)\langle n(X), s_1\rangle\} \\ I_2(p) = \max(\{0, \rho(X)\langle n(X), s_2\rangle\} \\ \dots \\ I_M(p) = \max\{0, \rho(X)\langle n(X), s_M\rangle\} \end{cases} \tag{2.15}$$

System $\mathcal{P}$ contains M equations and four unknowns at first sight: the three coordinates of the normal and the albedo. However, the simple rewriting $m(X) = \rho(X)n(X)$ allows us to reduce these unknowns to three since:

$$n(X) = \frac{m(X)}{\|m(X)\|} \tag{2.16}$$

$$\rho(X) = \|m(X)\| \tag{2.17}$$

So, for each pixel p, we can write a linear matrix system:

$$\mathbf{S}\, m(X) = \mathbf{I}(p) \tag{2.18}$$

with:

$$\mathbf{S} = \begin{bmatrix} s_1 & s_2 & \dots & s_M \end{bmatrix}^\top \qquad \text{and} \qquad \mathbf{I}(p) = \begin{bmatrix} I_1(p) \\ I_2(p) \\ \dots \\ I_M(p) \end{bmatrix} \tag{2.19}$$

System (2.18) must be solved in $m(X)$, in order to obtain the normal and albedo values. The max operator has been ignored in this system, as it is assumed that for a given pixel, if certain illuminations cause the observed colour to be black, these are ignored to ensure the resolution of the system not to be biased by inconsistent values.

This system admits a unique solution if at least three non-coplanar directional lights are used. This very basic version of PS can be easily improved with the use of neighbourhood schemes [103] to help introduce spatial regularisation into the estimation of normals. This means that the normals calculated for neighbouring pixels are encouraged to be similar, thereby reducing noise and measurement errors, and improving consistency for the reconstructed surface.

Other approaches such as rank reduction [128] are used to reduce the influence of outliers. More recently, deep learning-based methods were shown to overcome such classic approaches in terms of robustness [48].

Normal integration

For each pixel p, we can compute a unique $n(X)$ from (2.18) and (2.16), which we will denote $n(p)$. Consider $\mathcal{S}$, the parametric surface to infer from these normals. Its shape will correspond to a graph resulting from the 2D pixel structure:

$$p \longmapsto z = \mathcal{S}(p) \tag{2.20}$$

Let us denote:

$$n(p) = \begin{bmatrix} n_1(p) \\ n_2(p) \\ n_3(p) \end{bmatrix} \tag{2.21}$$

In addition, $n(p)$ can be derived from $\mathcal{S}$ under the orthographic assumption as follows:

$$n(p) = \frac{1}{\sqrt{\|\nabla \mathcal{S}(p)\|^2 + 1}} \begin{bmatrix} -\nabla \mathcal{S}(p) \\ 1 \end{bmatrix} \tag{2.22}$$

From Equations (2.21) and (2.22), we deduce:

$$\nabla \mathcal{S}(p) = -\left[\frac{n_1(p)}{n_3(p)}, \frac{n_2(p)}{n_3(p)}\right]^\top \tag{2.23}$$

Equation (2.23) shows a connection between the gradient of surface $\mathcal{S}$ and the normal values. This relation can be exploited to recover surface $\mathcal{S}$ by integrating its gradient $\nabla \mathcal{S}$. However, it requires the knowledge of the depth of at least one pixel p_0 to integrate $\nabla \mathcal{S}$. Then, according to [46], $\mathcal{S}(p)$ is obtained as:

$$\mathcal{S}(p) = \mathcal{S}(p_0) + \int_\lambda \nabla \mathcal{S}(p) \cdot dp \tag{2.24}$$

with λ a path from p_0 to p. Another approach to obtain $\mathcal{S}$ is to solve the following minimisation problem:

$$\min_{\mathcal{S}} \int_\Omega \left\| \nabla \mathcal{S} + \left[\frac{n_1(p)}{n_3(p)}, \frac{n_2(p)}{n_3(p)}\right]^\top \right\|^2 + \text{ boundary conditions} \tag{2.25}$$

This rewriting opened the doors of other techniques using variational methods [102, 70, 1], which managed to enhance the quality of surface $\mathcal{S}$ by using robust estimators and finite difference approximations of the gradients.

This section has provided us with all the knowledge required to understand how classic image-based methods work for 3D reconstruction. In the next section, we will discuss recent techniques based on a combination of classic approaches and neural networks.

2.2 Neural 3D reconstruction

Subsections 2.1.5 and 2.1.6 gave us an understanding of the classic approaches for 3D reconstruction methods still used today, with software such as Meshroom [38] or Colmap [27]. However, since the arrival of AlexNet in 2012, the research world has been captivated by neural networks and new neural methods for 3D reconstruction have emerged.

We have seen the emergence of generic methods attempting to reconstruct in 3D by learning from large synthetic databases [21, 140] showing low frequencies reconstructions. But also some direct alternatives to classic methods with the use of feature matching [63], depth map fusion [29, 105], end-to-end MVS pipelines [66, 122, 134] and structure-from-motion [112, 55, 51]. At the same time, we saw significant advancement in differentiable rendering, including the development of differentiable rasterizers [25, 36, 69] and differentiable ray tracing methods [95, 65, 110]. In addition to this work on 3D reconstruction, researchers also got interested in methods for the synthesis of new views [83, 45, 54].

All these recent developments used different 3D differentiable representations such as 3D occupancy fields [37, 80, 96], signed distance functions [49, 98, 122], but also density fields [83], point clouds [136], and triangle meshes [25].

2.2.1 Inverse rendering loop

A large panel of methods appeared after the 2010s, and most of them are what we might call "optimisation techniques based on an inverse rendering loop". They aim to deduce the scene's properties, such as geometry, materials and lighting, from images. As shown in Figure 2.5, the idea is to propose a fully differentiable rendering pipeline. It contains a scene representation and cameras that can be used by a differentiable renderer to produce rendered images of the scene at a state t. Then, a comparison between the input images and the rendered images (made with a similarity loss) is computed. Finally, the gradient of this loss is back-propagated into the scene representation to optimise it and create a more accurate representation.

It is therefore necessary to have a representation model that allows us to approximate the real images as closely as possible. Otherwise, there is a risk of converging towards a solution which, in certain cases, will not be able to explain the images and will result in aberrant data in the reconstruction (noise, breaks, etc.).

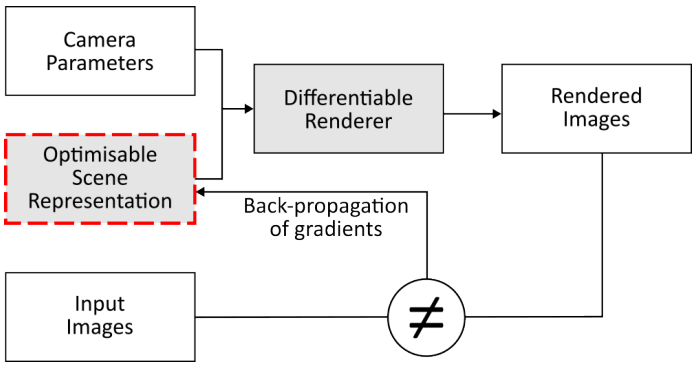


Figure 2.5: Inverse rendering loop. The scene representation with the camera parameters are used by the renderer to generate images of the scene. These rendered images are then compared to the input images to finally optimise the scene representation by back-propagating the gradient of the loss.

2.2.2 NeRF

In 2020, a novel approach called NeRF [83] changed the game for 3D neural reconstruction. Thousands of papers based on this method have been released in the last three years, showing the importance of this approach.

NeRF is a neural novel view method, which is based on an inverse rendering loop. It contains two key elements: a representation of the scene in the form of a radiance field and the use of a differentiable volume rendering engine, as shown in Figure 2.6.

The neural radiance field represented as a multi-layer perceptron (MLP), is a continuous 5D function that predicts a colour radiance $L_e \in \mathbb{R}^3$ as well as a density $\sigma \in \mathbb{R}$ at each point in space according to a given viewing direction d.

Considering a camera of optical centre o, the volumetric rendering of this view requires a per-pixel rendering. To do this, according to (2.6), we call $\tilde{X}_c$ the back-projection of a pixel p in the camera coordinate system. The viewing direction of this pixel is given by $d = \frac{\tilde{X}_c - o}{\|\tilde{X}_c - o\|}$. The corresponding back-projected

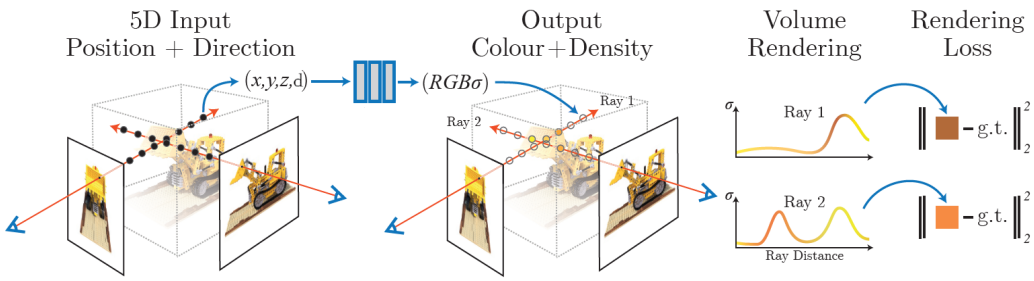


Figure 2.6: Figure from [83], reflecting an overview of the NeRF pipeline. We can notice the sampled point along the back-projected rays and a visualisation of the predicted colour $C(r)$ using the volume rendering.

ray r is therefore described by the parametric equation $r(t) = o + td$, where $t \in \mathbb{R}^+$.

The volume rendering equation tells us that the predicted colour $C(r)$ can be expressed as follows:

$$C(r) = \int_0^{+\infty} T(t)\sigma(r(t))L_e(r(t), d)dt \tag{2.26}$$

with $\sigma(r(t))$ representing a kind of probability (with no upper bound) of the presence of information (particles) at the position $r(t)$, and $T(t)$ the transmittance at position $r(t)$:

$$T(t) = \exp\left\{-\int_0^t \sigma(r(j))dj\right\} \tag{2.27}$$

This transmittance along the ray starts with a value of 1, but the more the ray comes across particles with high densities, the more the transmittance decreases, which explains why the following elements cannot be seen.

The next part consists of converting the integral into a sum, by introducing a sampling along the ray r. An upper bound t_{sup} will replace $+\infty$ and K points between 0 and t_{sup} named t_i, with $i \in \{1...K\}$, are chosen to replace the integral. NeRF chose to use the quadrature rule by [78] in its rewriting of the integral as the following discrete sum:

$$C(r) = \sum_{i=1}^{K} w(r(t_i))L_e(r(t_i), d) = \sum_{i=1}^{K} T(i)\alpha_i L_e(r(t_i), d) \tag{2.28}$$

$$T(t) = \prod_{j=1}^{i-1}(1 - \alpha_j) \tag{2.29}$$

with α_i defined as the discrete opacity and δ_i as the difference between two adjacent samples defined as follows:

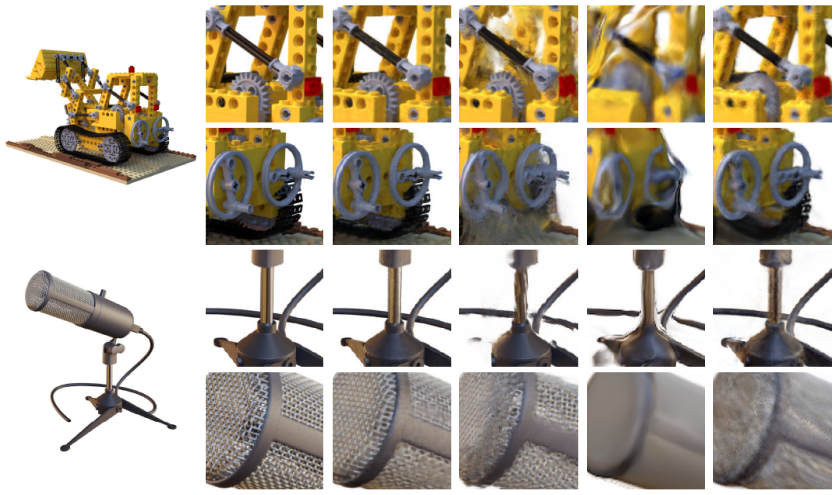
$$\alpha_i = 1 - \exp\{-\sigma(r(t_i))\delta_i\} \tag{2.30}$$

$$\delta_i = t_{i+1} - t_i \tag{2.31}$$

The choice of t_i samples is important to focus on high-density information, as explained in the NeRF paper but is not detailed further in this thesis. Once the rendering formula is determined, the remaining step consists of setting up a loss to optimise the system. The decision was made in favour of a colourimetric loss between the colour prediction $C(r_j)$ and the ground truth colour $\hat{C}(r_j)$ on all rays r_j, with $j \in \{1...L\}$:

$$\mathcal{L} = \sum_{j=1}^{L} \left\| C(r_j) - \hat{C}(r_j) \right\| \tag{2.32}$$

This method gives efficient predictions of new views, as shown in Figure 2.7, but suffers from a few problems: 3D models are noisy and the computation time is high (8-16 hours for a single optimisation).



GT NeRF [83] LLFF [82] SRN [111] NV [71]

Figure 2.7: NeRF novel view synthesis comparison (illustration from [83]). NeRF neural radiance model has learned the entire scene and is capable of providing high-quality new views compared with other methods.

NeRF can provide 3D reconstruction meshes using density data, but as density does not produce strong enough gradients at surface level, marching cube methods [73] have difficulty producing qualitative 3D surfaces, ending up with

irregularities and inaccuracies as shown in Figure 2.8. To fix this issue, researchers decided to model the geometry with a surface representation. This is the position taken by NeuS [122], which is also depicted in Figure 2.8, showing improved surface quality.

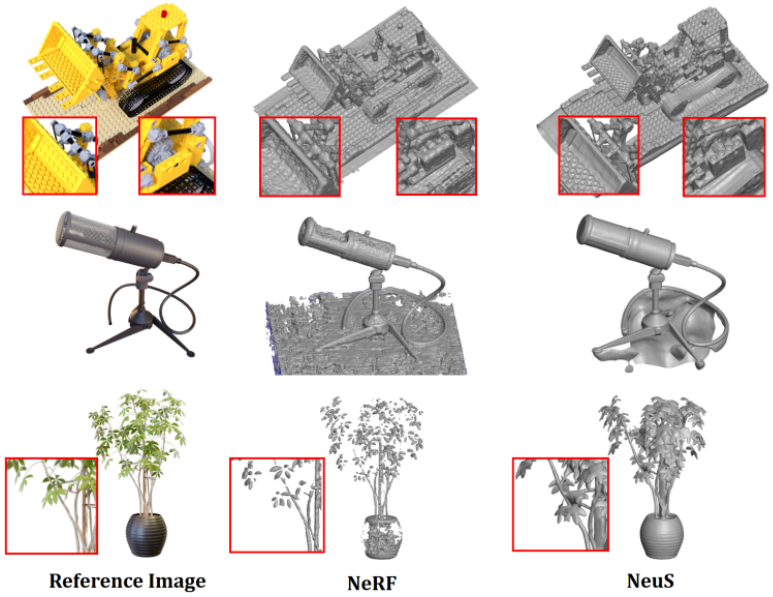


Figure 2.8: NeRF and NeuS 3D reconstructions from marching cube (illustration from [124]). The surfaces defined per NeRF are of insufficient quality, contain holes and do not specifically reflect the geometry. NeuS offers much cleaner and qualitative reconstructions, even if there are a few misunderstandings like the microphone base.

2.2.3 NeuS

NeuS [122], published one year after NeRF [83], proposes to keep the volume rendering of NeRF, but to modify the radiance field approach by splitting it into two distinct networks: a representation of the radiance L_e and a representation of the scene in the form of a signed distance function $f : \mathbb{R}^3 \to \mathbb{R}$. Thus, the surface of the object is represented as the zero level-set of the neural SDF, such that:

$$\mathcal{S} = \{x \in \mathbb{R}^3 | \, f(x) = 0\} \tag{2.33}$$

All NeuS work is therefore based on the function(s) that ensure the SDF information is transformed into a density in a differentiable way to make the volume rendering still workable.

NeuS introduces a probability density function $\phi_s(f(x))$, named S-density, composed of:

$$\phi_s(x) = \frac{se^{-sx}}{(1+e^{-sx})^2} \tag{2.34}$$

known as the logistic density distribution and as the derivative of the sigmoid function:

$$\Phi_s(x) = \frac{1}{1+e^{-sx}} \tag{2.35}$$

Combining both these functions, NeuS chose to rewrite Equation (2.30) as follows:

$$\alpha_i = \max\left\{\frac{\Phi_s(f(r(t_i))) - \Phi_s(f(r(t_{i+1})))}{\Phi_s(f(r(t_i)))}, 0\right\} \tag{2.36}$$

This rewriting allows an unbiased weight function $w(t_i)$ which reaches a local maximum at the intersection with the surface, i.e. where the SDF vanishes, to ensure an accurate representation of the surface. It is also sensitive to occlusions, explaining that points closer to the surface must contribute more than points further away along the same ray, even if their SDF are identical.

The final step consists of the addition of a second loss term to enforce the MLP representation to mimic a signed distance function. The main characteristic of an SDF is its gradient, whose norm is equal to 1 at almost any point in space (i.e. skeleton). An eikonal loss term, exploiting this property over the samples t_i of the rays r_j, for $i \in \{1...K\}$ and $j \in \{1...L\}$, was then added as follows to constrain the MLP:

$$\mathcal{L}_{\text{eik}} = \frac{1}{KL}\sum_i \sum_j \left(\|\nabla(f(t_{i,j})\| - 1\right)^2 \tag{2.37}$$

Among the many works that have resulted from NeuS, [129, 123] propose algorithmic optimisation to reduce the computation time, [50, 72] perform reconstructions with a limited amount of views, [24] works on complex reflectance materials, [124] on fine structures, and many others.

Publications show methods that are more and more open to different image modalities, using normal maps and depths, with research increasingly focused on reconstructing details, very large scenes and understanding complex material.

All the methods mentioned above are based on MVS. However, there are also rendering loop methods based on PS or multi-view PS data [133, 57, 56, 140, 19]. These approaches are currently the best in understanding complex reflectances, and in acquiring fine details [14].

Recent work using Gaussian splatting [60] overturned NeRF-based methods on novel view synthesis, introducing optimisations in reasonable time and the creation of novel views in real-time (up to 600 FPS). In addition to this, the recent work of Guédon et al. [39] provided a 3D mesh explanation of the optimized Gaussians, opening new doors for the future of 3D reconstruction.

This section showed the importance of new methods using neural networks for 3D reconstruction and demonstrated that they should no longer be neglected and be applied in symbiosis with the classic methods to constrain the neural 3D representations.

2.3 When 3D reconstruction meets refraction

Refraction has been extensively studied in the field of rendering due to its significant impact on producing realistic images, especially when dealing with elements like windows, water and caustics. The field of 3D reconstruction has predominantly focused on underwater scenarios involving air/glass/water interfaces. However, there is a notable lack of research addressing situations similar to ours, which involve objects inside a refractive medium.

2.3.1 Existing work

The first studies on refraction/reflection focused on the deformations linked to lenses and thin refractive layers, which were realised by Maas et al. [77] in the case of applying a fine layer of glass between air and water.

Much subsequent work has focused on data acquisition and reconstruction of underwater scenes, with some work on image correction that used epipolar geometry [75, 53, 52], or those by [2, 130] using neural networks. Some approaches are centred on the use of light field cameras [47, 137] to analyse light directions to cancel refraction and predict refractive index. Morris et al. [84, 85] have focused on surface and depth predictions in the context of air/water interfaces using refracted patterns. Chen et al. [23] worked on the projection of fringes onto the refractive surface to understand its geometry.

For underwater reconstruction, there have also been significant contributions based on photometric stereo methods with [120, 90, 88] that address the challenges posed by light absorption in water. These studies consider factors such as the loss of colour at greater depths and light absorption by unclear water, which impacts accuracy and quality of the reconstructed images. Narasimhan et al. [92] studied refraction in photometric stereo in the context of a fronto-parallel plane interface between two media, modelling the different intensities of light rays and their transformation as they pass through the refractive interface. Fan et al. [31] have carried out similar studies using laser triangulation.

Going back to theoretical studies, the work of Sturm [114] on camera models for structure-from-motion, including those with refracted axes, is particularly noteworthy. A few years later, Chari and Sturm proposed in [22] a rewriting of the epipolar geometry for the case of refractive planar interfaces between the scene and the camera. In the same vein, Haner et al. [40] propose a wider method for estimating camera poses in the case of planar interfaces.

Li et al. [61], as well as [132], were interested in single-view 3D reconstruction using bi-prisms, providing two views of the scene from different angles in the same image, unlocking access to photometric stereo methods. Chen [26] and Gao [33] focused on the rotation of refractive panels between the camera and the scene to provide multi-view information. Others, such as [139, 94], have preferred to use mirrors in the scene to obtain additional information. Recent work [3] has even gone so far as to use iris reflection to reconstruct 3D scenes.

Other projects [89, 126, 18] referred under the term bathymetry (i.e. the measurement of ocean depths and relief) also kept interest in refraction.

Recently, with the use of differentiable rendering, some work on the reconstruction of transparent objects has appeared. For example, [9] attempted to fit surface models onto images distorted by refraction in the case of very basic geometry (cube, sphere). Lyu et al. [76] proposed an approach mixing multiview and patterns to recover a triangular mesh. Li et al. [67] mixed a neural approach and a rendering applying Snell's laws, and Shao [109] studied the understanding of polarization for 3D reconstruction.

Finally, we find the work of Xion et al. [131] modelling the surface of waves over time to reconstruct the seabed, and Tong et al. [117] proposing in 2023 a neural multiview approach to reconstruct objects trapped inside cuboids with refractive properties.

2.3.2 Contributions

The contributions of this thesis will refer to the adaptation to refraction of the different approaches presented in this chapter. We will start in Chapter 3 with a presentation of our work on a method based on shape-from-silhouette allowing both the calibration of extrinsic camera parameters and the 3D reconstruction of polyhedra. Chapter 4 will focus on a multi-view refracted reconstruction method for providing 3D point clouds of objects trapped in polyhedral shapes and arbitrary shapes. Chapter 5 will concern the implementation of a refracted PS method in an infinite plane framework. Chapter 6 will be a bridge between classic and neural methods, with a work based on NeuS [122] using multi-view PS (MVPS) data, proposing a re-parametrisation

of the input data in the form of normal and reflectance maps to generate 3D reconstructions with numerous fine details. Chapter 7 will discuss current work on a two-stage method using refracted MVPS data to compute and integrate refracted normal maps. Finally, the last chapter will discuss the limitations and perspectives of this work to conclude this 3-year project.

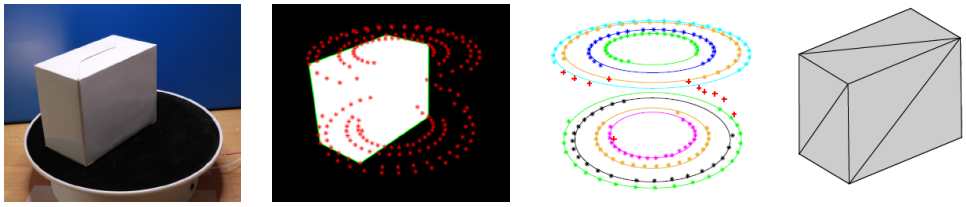
Chapter 3

3D reconstruction of convex polyhedra using shape-from-silhouette

As explained in the introduction, we began our work on insects imprisoned in transparent white resin cuboids with a refractive index of 1.56, close to the 1.54 of amber. This led naturally to the idea of modelling a cuboid/polyhedron in a scene using a shape a priori. Moreover, this method can also be used to calibrate external camera parameters. This work was published at the Quality Control by Artificial Vision (QCAV) conference and is entitled: "A Shape-from-silhouette Method for 3D Reconstruction of a Convex Polyhedron"[15].

This method, not limited to cuboids but also polyhedrons, represents an accurate and robust pipeline for the 3D reconstruction of convex polyhedral objects based on multi-view data obtained with a circular motion (turntable). The first step consists of detecting the corners of the polyhedron, and then mapping them into elliptical trajectories. These trajectories can then be interpreted to provide a triangular mesh and external parameters for the cameras without any use of markers. Detecting the corners of polyhedra and mapping them to the elliptical trajectories they explain, not only provides us with an easy-to-use method that requires no markers but also a triangular mesh of the polyhedron and calibration of the cameras' internal parameters, as shown in Figure 3.1.

As polyhedron corner detection is silhouette-based, this method is robust to complex/textureless materials. This is, in general, no the case for structure-from-motion methods. Note however that concave cases are impossible to reconstruct because of silhouette-based corner detection. In addition, processing polyhedra with more than 10 faces can lead to errors. Indeed, the



(a) (b) (c) (d)

Figure 3.1: Overview of our method: (a) Input image of a box on a turntable; (b) Segmentation of the box and corner detection for all views; (c) Estimation of elliptical trajectories; (d) Reconstructed triangular mesh of the box.

corner detection can fail due to the increase of obtuse angles. Moreover, elliptical trajectories will become increasingly more difficult to detect because of the overlapping between the corners. Though it is possible to robustify this method, for the application case of insects in resin cuboids, this is perfectly sufficient.

Chapter 4

Refracted multi-view stereo

In the previous chapter, we obtained calibrated cameras and a triangular mesh representation of the refractive medium supposedly polyhedral. With this knowledge, we can now turn our attention to reconstructing the object inside it. In this context, which is very suitable for MVS methods, we will add our knowledge regarding the laws of refraction to propose a refracted MVS pipeline as seen in Subsection 2.1.5 to reconstruct the insect inside the refractive medium.

This is how we led to the paper entitled "Multi-view stereo of an object immersed in a refractive medium" [16], proposing a complete 3D reconstruction pipeline based on refracted multi-view data. This pipeline is an extension of the method of Brument et al. [15] and the work of Cassidy et al. [20] on refracted stereo multi-view reconstruction. The main idea is to model the light paths according to Snell's equations for projection and back projection through the medium within the MVS method.

This article also includes work to refine and robustify the method in [20] by adding reconstruction of polyhedral media with more complex geometry, as shown in Figure 4.1, as well as studies on the determination of the refractive index of the medium. Finally, we also studied the consequences of the usage of different levels of precision in the definition of the shape of the medium (number of faces).

The main constraint of this method is the computation time: restricted to polyhedral refractive media, the results can be obtained within 10 to 24 hours, depending on the precision required. In a time frame, due to the complexity of the number of faces and despite the optimisations implemented to process only the faces required during optimisation, this time evolves from one day to one week.

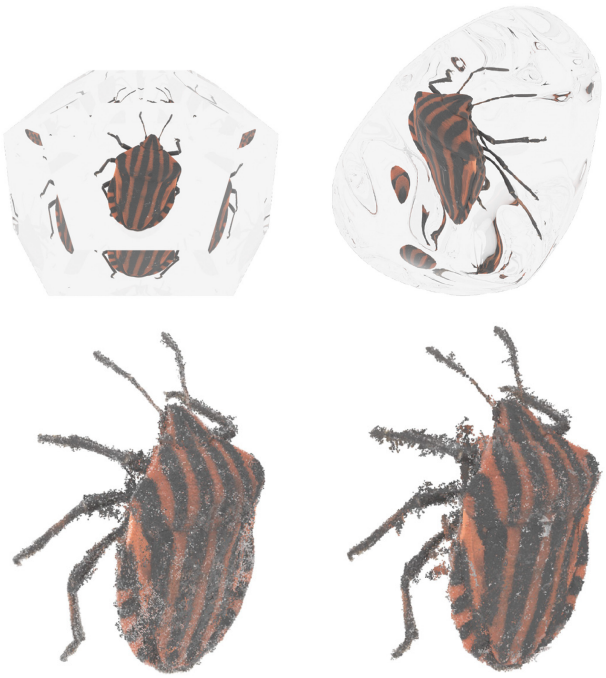


Figure 4.1: First row: Examples of input images of a graphosoma inside a dodecahedron and inside a refractive medium of arbitrary shape. Second row: Corresponding 3D reconstructions using our method.

It is also worth noting that just as we were finishing fine-tuning this method, neural applications such as Tong et al. [117] appeared, allowing in the case of a cuboid to model the surface insect representation with superior quality. The unavailability of the code prevented us from comparing our method with their, but it encouraged us to explore neural methods such as [83, 122] for our future work, rather than seeking to optimise this pipeline even further.

Chapter 5

Refractive photometric stereo

Chapter 4 highlighted that our adaptation of a geometric method (MVS) to refraction has shown positive results when it comes to the use of Snell's laws to adapt 3D reconstruction methods. However, we must not forget a second important family for 3D reconstruction, as seen in Table 2.1: photometric methods. Fan et al. [31] explored the case of an underwater camera, with refraction caused by the encasing of the camera. In this work, we focus on a different situation: a refractive photometric stereo method with an (infinite) planar refractive interface.

In this contribution, entitled "On Photometric Stereo in the Presence of a Refractive Interface" [101], the aim is to adapt the model seen in Subsection 2.1.6 to refraction by taking into account an infinite plane separating two homogeneous media. The difference with the work of [31] lies in the use of a plane that is not necessarily fronto-parallel, in the presence of an orthographic camera model instead of a perspective one, and in the absence of a laser to calibrate depth.

This article shows how to modify the directions of both the camera and the directional lights according to Snell's laws, as well as the intensity of the lights by applying Fresnel's laws. Following these principles, we can create a bijection with a new underwater scene, this time without refraction, enabling us to solve a classic photometric stereo problem, as shown in Figure 5.1.

Our results show that taking refraction into account provides higher-quality 3D reconstructions, while ignoring it leads to much flatter results. We also point out that the depth ambiguity prevents us from using Beer-Lambert's law (see Equation (1.7)), because of the unknown length l, causing the impossibility of modelling the colour absorbance in our model.

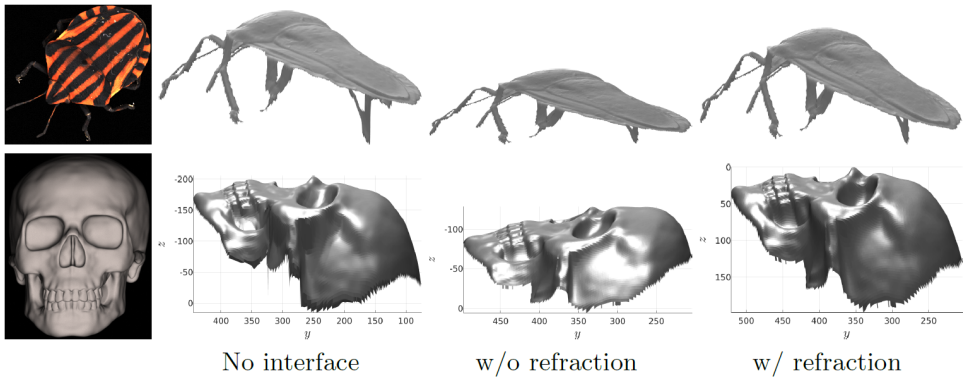


Figure 5.1: Photometric stereo 3D reconstructions on synthetic data. From left to right: In air (no interface), and in water (air/water interface), without and with refraction usage for the reconstruction.

Chapter 6

Reflectance and normal-based neural inverse rendering

Around 2020, as we saw in the introduction, neural inverse rendering methods have exploded, including approaches involving 3D reconstruction. NeuS [122] emerged at this time as a very convenient method for understanding the logic applied with volumetric rendering. While learning how it works, and modifying its code to make our version of ReNeuS [117] (original code is still unavailable at the time of writing this thesis), we worked on its adaptation to MVPS data to exploit the best of photometric and geometric methods to reconstruct objects with high fidelity on low and high frequencies. However, this method in not based on refraction.

This contribution to general 3D reconstruction, based on NeuS, is called "RNb-NeuS: Reflectance and Normal-based Multi-View 3D Reconstruction" [14]. Our initial insight emerged from recognizing that existing methods using this type of data were inadequate at leveraging multi-illumination data to recover fine details. After considering an excess of parameters for optimisation, which often did not accurately reflect the real data, we developed a single-objective method.

The first phase consists of using any PS method to obtain normal and reflectance maps for each view. With these new inputs, we proposed a per-pixel re-parameterization to render MVPS images mixing normal, reflectance maps and per-pixel optimal illumination. Our method is based on simple reflectance modelling focusing purely on albedo, but it can be adapted to more complex models. The joint optimisation of normals as a gradient of the neural SDF and albedo as a prediction of the colour network (independent of the viewing direction) has enabled us to recover many details in the 3D reconstructions, as shown in Figure 6.1.

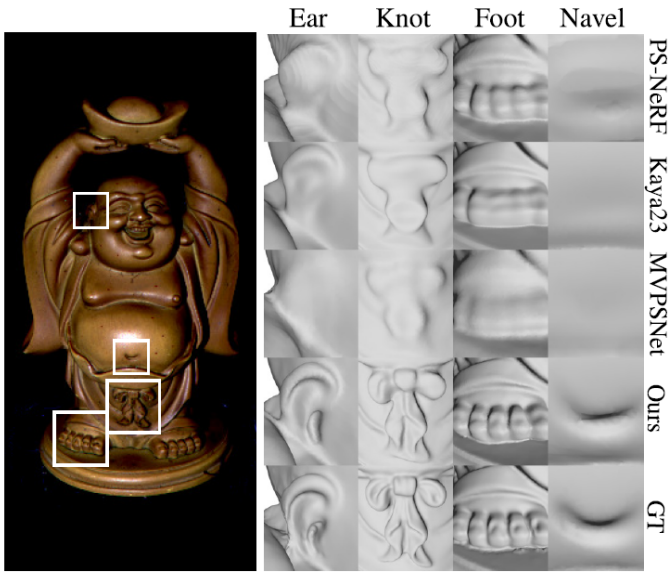


Figure 6.1: Comparison between our method and state-of-the-art approaches on the Buddha from the DiLiGenT-MV dataset [64] .

A major limitation of this method is its optimisation time: each scene optimisation can take between 6 and 15 hours, depending on the graphics card. To compensate for this, we have adapted our method to NeuS2 [123], a new version of NeuS based mainly on CUDA, optimising scenes in 5 minutes. However, especially for new methods, this backbone transition is not free of charge, as the absence of an automatic differentiator like Pytorch requires manual calculation of the gradients to be propagated in the code for any modification.

This knowledge of neural methods and this observation of the 3D reconstructed details using MVPS data encouraged the work in the next chapter, this time based on refraction.

Chapter 7

Refractive normal acquisition and integration

In Chapter 6, we proposed a 3D reconstruction method based on MVPS data. This work provided us with sufficient knowledge to start adapting the neural approaches to refraction. The idea is to set up a pipeline based on images of an object inside a refracting cuboid under multi-view and multi-illumination. For this refractive MVPS approach, we propose a two-step pipeline. The first stage involves the understanding of illumination propagation inside a cuboid shape to generate refracted normal maps. The second step requires integrating these normal maps using multiple views. This step adapts the NeuS volumetric approach to estimating normals instead of transmittance while considering refraction.

To implement this pipeline, we assume the same data as before: perspective cameras and calibrated directional lights. The object to be reconstructed will be assumed to be Lambertian, and the interface surrounding it a known cuboid shape of constant refractive index and neutral white colour. Contrarily to the previous chapters which all appeared in published paper, the results will be provided only on synthetic data, the method still having to be evaluated on real data.

7.1 Obtaining refractive normal maps

Input data This step consists of estimating the normal map for each view $i \in \{1...N\}$, independently. Each view is calibrated and captured under M varying directional illuminations s_j, $j \in \{1...M\}$. The binary masks B_i of each view of the object are also needed, as well as the refractive cuboid medium mesh, expressed in the world coordinate system, and its index of refraction.

As seen in Subsection 2.1.6, under the Lambertian assumption, to obtain the normal $n(p)$ corresponding to a given pixel p, photometric stereo tells us that we need at least 3 non-coplanar illuminations. Following Equation (2.19), this normal is obtained by estimating the vector $m(p)$ that best solves the linear system $\mathbf{S}\, m(p) = \mathbf{I}(p)$. Therein, vector $\mathbf{I}(p)$ contains the input greyscale values at pixel p, and matrix S contains the illuminations, which must be carefully modeled to account for refraction.

Our work in [101] showed that in the case of an infinite planar interface between the camera and the object to reconstruct, it is possible, in a directional multi-illumination framework, to obtain the normal $n(p)$ at each pixel p by solving the same linear system as before, by applying Snell's and Fresnel's laws. However, with a cuboid representation, geometry causes light contributions at a pixel to be far more complex than a refracted direction and Fresnel-affected intensity.

Figure 7.1 shows a 2D case with a square medium and a circle representing the object to be reconstructed. In this figure, we have simulated the propagation of a single directional light, and we can see that not all areas of the object are illuminated in the same way. Indeed, some parts of the circle will receive direct refracted light, but some other parts will also receive additional light contributions due to reflections/bounces in the cuboid. Figure 7.2 shows the light distribution when the light enters from two edges. As expected, the light contribution is even more complicated to predict. In the case of a cube, the light would enter most of the time from three different faces and would bounce more than twice.

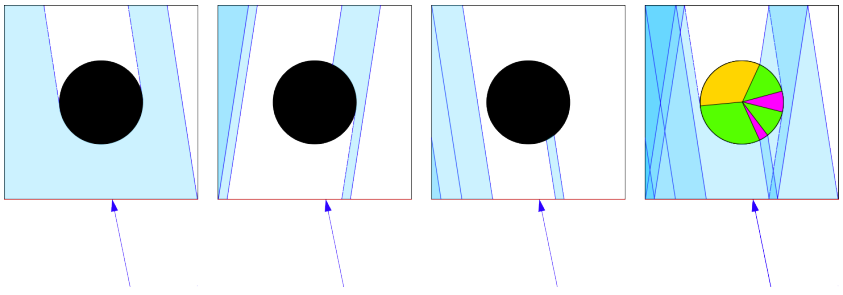


Figure 7.1: 2D simulation of rays in a refractive square with a circle object inside. The light enters from the bottom edge. We can see, from left to right, the different bounces inside the medium, and on the right, the final lights by summing all contributions. The colours **orange**, **green** and **magenta** correspond to the distribution on the circle of areas hit by 0, 1 or 2 different illuminations.

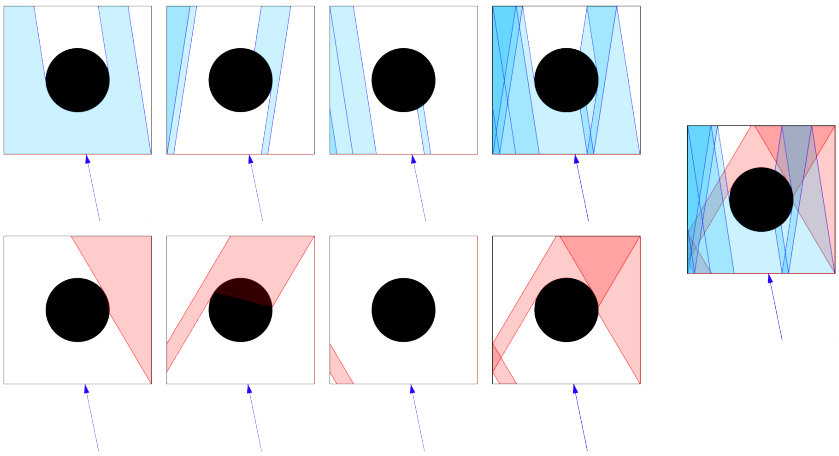


Figure 7.2: Same simulation as in Figure 7.1, but with a second row showing the simulation with the light propagation from a second edge of the square, and finally the expected light distribution after only 2 bounces.

From these simulations, we realised that we would need a simulation of the light contributions in the cuboid, as well as an a priori of the back-projection of each pixel in the medium, to predict the light contributions.

For each known light direction, we simulated its propagation through the medium, ignoring the internal object to simplify the modelling of intersections and to store the light bounces more easily. With our knowledge of the cuboid and the preliminary 3D reconstruction, we obtained an approximate 3D position for each pixel. For a given pixel p and its corresponding 3D point X, we can query the ray simulator to identify all the light rays arriving at this point, denoted as $Q(X)$. Next, we eliminate all rays that cannot physically reach that point due to the estimated 3D reconstruction, with $RQ(X)$ representing these removed light contributions. This involves reversing the path of all these rays and ensuring they do not intersect the object, as illustrated in Figure 7.3. Once this validation is complete, we sum the remaining light contributions, weighted by their intensities α (which vary according to Fresnel's laws), and use in the linear system this new light $s(X)^*$:

$$s(X)^* = \sum_{(s_i,\alpha_i)\in Q(X)} \alpha_i s_i - \sum_{(s_j,\alpha_j)\in RQ(X)} \alpha_j s_j \tag{7.1}$$

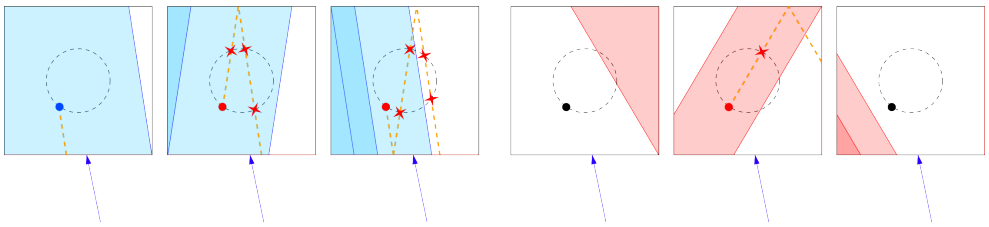


Figure 7.3: Simulation similar to that in Figure 7.2, ignoring the collisions with the circle. Looking at a specific point on the circle, we can see in the orange dashed line the path of light to reach this particular point on different levels of bounces. When the dot is **black**, no light is contributing. When it is **red**, a light reaches this position, but intersections with the object (as red crosses) are on the way, so this contribution has to be removed. Finally, when the dot is **blue**, light reaches the position without intersecting the object, adding a contribution to the real dot illumination.

7.2 Refractive normal integration

Multi-view normal integration consists of reconstructing the surface of an object by merging the normal maps from different points of view. Based on NeuS2 [123], we have built a neural normal integrator that we will use as a backbone for refraction.

NeuS2 was introduced as a fusion of InstantNGP [87] and NeuS [122]. It combines the optimised architecture of InstantNPG with the neural SDF (f) representation of the surface of the object to be reconstructed, allowing NeuS-like reconstruction in 5 minutes instead of 8-16 hours.

The optimisation is the same used by NeuS (see Equation (2.32)) including the same colour prediction (see Equation (2.28)). This architecture lends itself particularly well to multi-view reconstruction, however, it is not suitable for integrating normals. While remaining based on a volume rendering function, we have chosen to adapt its formula to predict a normal rather than a radiance. As our object is represented using f, we have changed the colour prediction term $L_e(r(t_i), d)$ by $n(r(t_i)) = \nabla f(r(t_i))$, which gives us the following normal prediction of a given ray r:

$$N(r) = \sum_{i=1}^{K} w(t_i) n(r(t_i)) = \sum_{i=1}^{K} T(i)\alpha_i \nabla f(r(t_i)) \tag{7.2}$$

With this new rewriting, the multi-view normal integration is possible. For our purpose, we would like to be able to apply it to refractive normal maps. At this stage, we expect the geometry of the medium and its refractive index

to be known. The required change, to take the medium into account, is to apply Snell's laws to the back-projected rays r used for volume rendering and to compute their refractive version $\hat{r}$. These new rays will replace the originals, allowing the reconstruction of a scene in a single, refraction-free environment.

One of the NeuS2 constraints is its need for analytical gradients in the code. Changing the loss term requires updating all the old gradients related to the colourimetric loss with the new ones related to our three losses:

$$\mathcal{L}_{\text{normal}} = \frac{1}{L}\sum_{j=1}^{L}\left\|N(\hat{r}_j) - \hat{N}(r_j)\right\|^2 \tag{7.3}$$

$$\mathcal{L}_{\text{eikonal}} = \frac{1}{KL}\sum_{i=1}^{K}\sum_{j=1}^{L}(\|\nabla f(\hat{r}_j(t_i))\| - 1)^2 \tag{7.4}$$

$$\mathcal{L}_{\text{mask}} = \frac{1}{L}\sum_{j=1}^{L}\left[-B(r_j)\log\left(\Phi\left(\overline{W}_j\right)\right) + (1 - B(r_j))\log\left(1 - \Phi\left(\overline{W}_j\right)\right)\right] \tag{7.5}$$

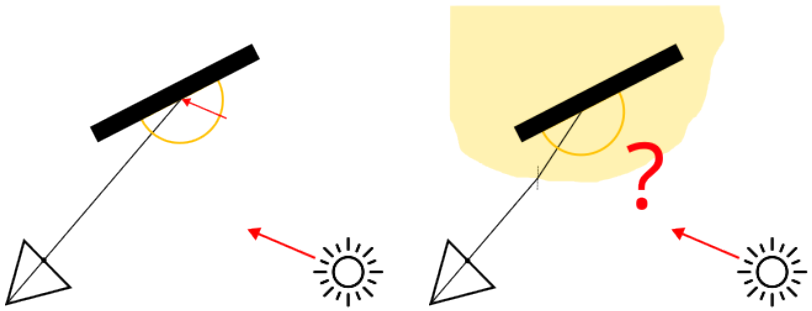
with:

$$\overline{W}_j = \sum_{i=1}^{K} w(\hat{r}_j(t_{i,j})) \tag{7.6}$$

7.3 Results and issues

The most traditional method for evaluating 3D reconstruction algorithms is to generate synthetic data, ensuring perfect camera calibration and the absence of noise in the data. Tools such as Blender, however, are unable to generate data using the refracted MVPS model, because of the null measurement phenomena it generates.

This phenomena can be understood as follows. In the path-tracing rendering process, as discussed in Subsection 2.1.2, one starts from the camera or the light source and then shoots numerous rays to define paths from the camera to the light, passing through the objects in the scene. Typically, as illustrated in Figure 7.4, when encountering a Lambertian object, the algorithm checks if the diffusion hemisphere includes the direction of the directional lighting. If it does, this direct lighting is accounted for. However, with a refractive medium, this calculation becomes very expensive because this requires finding the normal to the surface that accounts for a refracted direction similar to that of light, and then finding out if there is a similar normal on the medium that can be reached from the hemisphere. Since rendering engines do not perform



(a) (b)

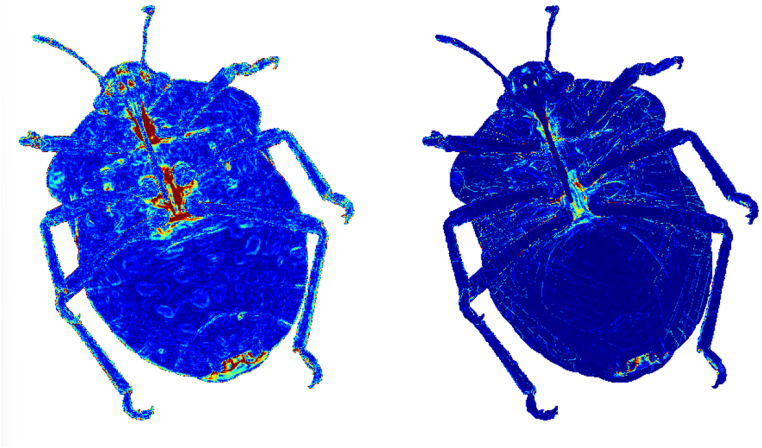
Figure 7.4: Illustration representing the light-object-camera path for path tracing with a Lambertian object. (a) In the absence of a refractive medium, the light direction is found inside the hemisphere around the normal at the surface. (b) In the presence of a refractive medium, the path of light from the object to the light is not straightforward and requires heavy optimisation computations.

this optimisation, they only shoot a lot of rays, but none of them emerge from the medium in a direction similar to that of light, which explains the zero measurement.

To validate our method, we thus opted to develop our own rendering engine, distinct from path tracing. This engine is based on simulating rays within the medium to determine light contributions at every point within it, thereby addressing the null measurement problem.

Once our synthetic data had been generated, we used a homemade version of [117] based on NeuS2 to do an initial 3D reconstruction. Using this initial reconstruction, we used our light ray simulator to compute the refracted illumination matrix $\mathbf{S}$, and subsequently estimate the refracted normal maps using photometric stereo. Figure 7.5 shows the mean angular error (MAE) between our normals and the ground-truth normals. We can see a reduction in the angular deviation between the normal maps obtained from the first ReNeuS2 estimation and those obtained with the MVPS data and the ray simulation.

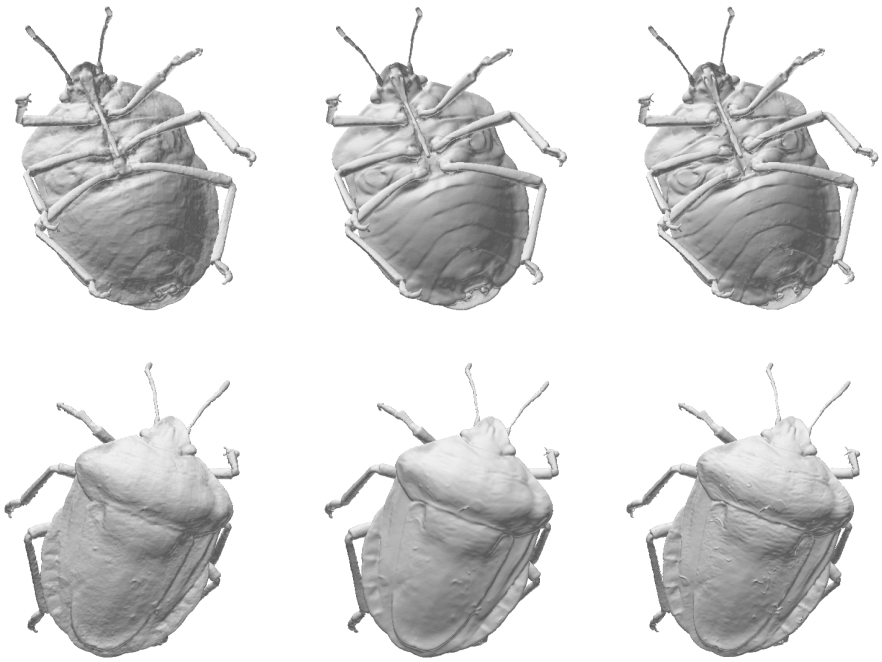
Figure 7.6 shows the difference in reconstruction between refractive MVS and refractive MVPS (after integration of the normals). We can see that our method provides smoother surfaces with more details, as well as a correction on some poorly reconstructed areas. However, concerning real data, our initial tests were inconclusive, as this method requires a high degree of precision when it comes to calibrating the various elements (cameras, lights, cuboids).



(a) (b)

Figure 7.5: Angular error on the normals for a graphosoma trapped inside a refractive cuboid, using: (a) ReNeuS2 from the reconstructed mesh; (b) Our approach.

Future acquisitions will focus on the improvement of the quality of the data, to prevent these issues and to enable us to validate our method or not.



(a) (b) (c)

Figure 7.6: Two 3D reconstructions of a graphosoma inside a cuboid (synthetic data), using: (a) ReNeuS2; (b) Our solution. The ground truth is shown in (c).

Chapter 8

General conclusion

8.1 Conclusion

As we have seen at the beginning of this thesis, this work is set within the context of exploring ways to provide 3D models of insects trapped in amber. However, three years is not enough time to propose a complete reconstruction pipeline to overcome all the complexities of amber. Considering that, we have chosen to focus on the major property of amber, which is the notion of reflection and refraction, described by Snell's and Fresnel's laws. Our contributions have shown how both classical and recent neural methods can be modified to suit the refractive scene. Our initial work involved using multi-view data to propose a method for estimating the shape of a polyhedral interface, and then using this new knowledge to adapt the multi-view stereo (MVS) method by applying Snell's laws for the back-projection and projection of light rays. This approach enabled us to understand that it was possible to produce 3D reconstructions that were faithful to the geometry of the insect, whatever the geometry of the medium. We also demonstrated that the algorithmic cost of the projection operations, which consist of finding the shortest path for the light to take, was directly related to the complexity of the shape of the medium.

Our work also focused on the adaptation of photometric stereo to refraction in an infinite planar interface (aquarium) framework. This contribution allowed us to show how the orthographic camera model, as well as the directional light model, were affected by refraction. We were able to observe a bijection in this specific case between the scene and a second scene involving a unique medium (air). Although the use of orthographic cameras provided a simplistic solution to the refraction problem, this solution remains rather limited in our context, as the depth ambiguity does not allow colour absorption to be taken into account, and mainly because the planar model is very far from the shape of amber.

The work of Tong et al. [117] in 2023 on a neural refracted MVS method subsequently allowed us to look at new neural methods. Indeed, by reproducing their method on the data from our classic refractive MVS method, we were able to notice the quality of the 3D reconstruction of the insects, no longer in the shape of a point cloud but in a surface shape. The second strong point of this method is that there is no need to solve the projection equation, which can considerably reduce computing time. However, Tong et al. did not generalise their approach to all types of shapes and restricted themselves to a cuboidal shape, which is much more acceptable for simulating rebounds in the medium. In future work, we will present a method adapted from theirs, using masks to get rid of luminous bounces and to be able to use much more diverse shapes of medium.

In order to familiarise ourselves with neural methods, we worked on a project involving the use of multi-view, multi-illumination data to reconstruct objects with attention to fine details. This contribution enabled us to understand how volumetric methods work for 3D reconstruction and to see the potential for using them to understand different materials. This contribution will be valuable for refraction, giving us access to a backbone to modify that can already understand the fine details of insects and possibly take complex materials into account.

Our latest work has focused on the integration of refracted normals within the context of a cuboid shape for the medium but with the idea of using polyhedra in the future. This contribution, which is still in progress, should enable us to better model the behaviour of light rays in a non-infinite medium. As seen in Chapter 7, light rays in a refractive medium will bounce in a manner that generates an almost infinite number of different light contributions with intensities that are initially close to the original intensity but become progressively smaller until they are infinitesimal. The approximation of this infinite sum of light contributions made in Chapter 7, which seemed accurate to us, is in fact slightly incorrect when we compare our results with real images. Indeed, we have chosen to ignore light contributions below a certain intensity level, but the sum of all these small contributions is likely non-negligible. It will be important, in future work, to be able to model the interaction of light in the medium more precisely, to exploit these photometric data and produce accurate 3D reconstructions.

In the end, the impact of this work was to demonstrate the feasibility of 3D reconstruction of objects trapped in a refractive medium. We were able to show that it was possible to adapt traditional methods, enabling us to obtain results that respected the geometry of the objects. We were also able to use the new neural methods to produce surface reconstructions of the same order

of quality as those in the air. We also need to keep in mind that the quality of 3D reconstructions involving refraction is very dependent on the calibration of the various components of the scene (cameras, lights and refractive medium), and that future work on both the macroscopic and microscopic scales will require a great deal of rigour when it comes to data acquisition.

This thesis has shown that the initial project of reconstructing insects in amber is certainly achievable. There are still many challenges ahead before we can achieve even the first microscopic-scale reconstructions, but current work on refraction, different materials, detail reconstruction and fine structures will eventually make it possible to reconstruct insects with a high degree of fidelity and then do the same for versions embedded in amber.

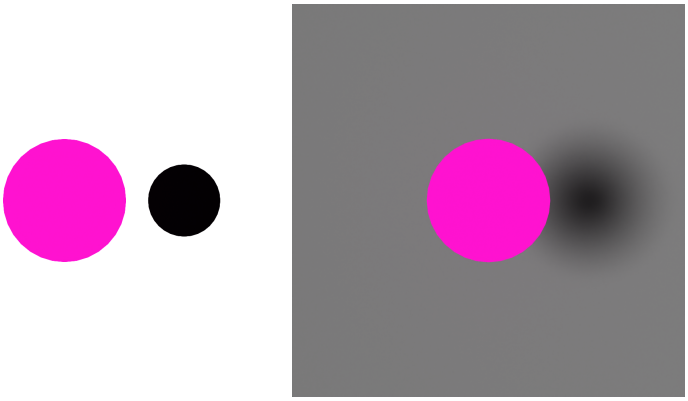
So, we should not be surprised if, in a few years, museums offer visitors the chance to see not only the dinosaurs of yesteryear, but also insects from those same eras, and to explain how they got here by following the almost eternal path of fossilisation.

8.2 Limitations

Despite the simple cuboid form, when dealing with refraction, we were confronted with several serious problems and limitations we will need to understand before progressing to more accurate amber models.

8.2.1 Synthetic data

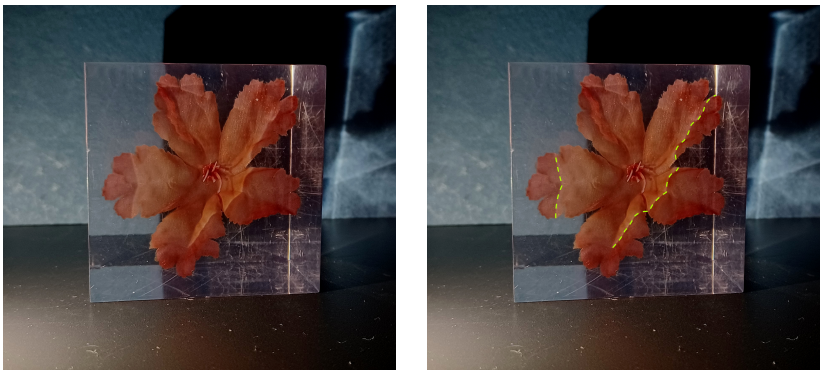
Synthetic data is essential for verifying that 3D reconstruction methods work properly under quasi-optimal conditions. As we saw earlier, these tools easily allow us to render realistic scenes with different materials, while maintaining colour consistency across the different views. However, creating PS data with these models becomes significantly more complex, especially when trying to follow the physical models. The most complex problem is adding the refractive medium to the scenes. The modelling of Snell's and Fresnel's laws is not always strictly respected because it can lead to zero measurements and prevent images from being generated. At the current time, the generation of MVPS data with refraction is impossible because it is too far away from reality. Normally, with directional lighting, a shadow cast by object A on object B produces a shadow with sharp edges. In the case of refraction, with B inside a refractive medium, the cast-shadow will be shifted due to the refracted rays, but the edge must remain sharp. However, looking at the difference between (a) and (b) in Figure 8.1, we can see that the shadow of the circle has shifted in (b), but the edges are no longer sharp and the luminosity has dropped drastically, which is not physically reliable.



(a) (b)

Figure 8.1: Rendering with Blender of a scene with a 45° azimut rotation directional light, a plane and a circle between the plane and the light. (a) The circle casts a sharp shadow on the plane. (b) When the plane is inside a refractive medium, as expected, the shadow is deviated due to refraction. However, for no reason, the shadow is no longer sharp and the brightness is much lower.

Figure 8.2 shows in the real case of a cuboid with a flower trapped inside, how the faces of the cuboid cast shadow on the flower, validating Blender incorrect renderings.



(a) (b)

Figure 8.2: (a) Image of a flower trapped inside an epoxy cuboid illuminated by a directional light. In (b), we emphasize the faces borders projected into the flower, which exhibit sharp boundaries.

Rendering engines are built for artistic purposes for the most part, which is why care must be taken when using them to produce physically realistic data. Taking a survey of the existing set of engines for future work could be useful to generate images that can be used for photometry.

8.2.2 Data acquisition

Acquiring data with refraction poses significant challenges due to the need for highly precise calibration. The complex paths between the camera, the object, and the light require accurate modelling of the medium: any imprecision can hinder the reconstruction of fine structures. Distortions in light direction or intensity can lead to inaccuracies in the light contributions after reflections, complicating image interpretation.

Another challenge is the acquisition of objects within amber or simpler models such as resin cuboids. These epoxy models are not abundant and often objects trapped inside do not follow the Lambertian assumption. The materials are more complex than expected and tend to break our naive models. Moreover, working with actual pieces of amber introduces additional difficulties. The amber pieces can vary from a few millimetres to several centimetres, with insects to be reconstructed at millimetric scale. This requires modifying the acquisition setup to include microscopic cameras and their related camera models. Furthermore, issues related to focus and calibration for these new cameras must be addressed.

8.2.3 Refraction/reflection complexity

Fresnel's laws explain that for high incident angles, the reflectance coefficient R will be significant, causing a white filter on the surface of the medium. This effect partially damages the images by causing them to lose sharpness. In real acquisition cases, we also observe specular points on the medium due to the lights, which deviate from the directional model. They behave essentially as punctual light sources, causing this detrimental effect on our data.

Although the modelling of reflection/refraction is relatively straightforward at first sight when dealing with a single interface, it becomes much more complex when bounces have to be taken into account, as seen in Chapter 7. We expected real data to fit our simulated model with the presence of a lot of light patterns projected on the object, in reality, looking at the actual data, we do not find this pattern of light bouncing off the surface of the object. Some sharp lines can be seen, but beyond that, the impression is more an ambient lighting as a blend of all the rebounds. This analysis of the behaviour of light in the event of bounces requires more in-depth work to be able to figure out how to better model the lights.

8.3 Future work

Future work will focus on implementing a faster neural refracted MVS method than the one proposed by [117], building on our work adapting NeuS2 or trying to adapt new methods such as SuperNormal [19] or those based on Gaussian splatting [39].

We would also like to continue our work on multi-view, multi-illumination data to gain a better understanding of lighting patterns with bounces, but also to exploit the information in the normals to produce real results. Knowing that methods using photometric stereo can be adapted to explain fairly complex reflectances, we would eventually like to go beyond albedo and model reflective, transparent and metallic surfaces. In the future, we are looking to set up a kit to acquire refractive MVS or refractive MVPS data as accurately as possible, and with as little effort as possible. In addition, a polarising filter could be added during image capture to eliminate reflections on the faces of the medium. As well as simplifying the model a little, this would probably improve the sharpness of the images.

Future work will then focus on gradually adding the elements seen in Section 1.2, such as taking account of colour/absorption. Then, when we are able to acquire real data, bubbles and dust will have to be taken into account. Finally, attention can be paid to the smallness of the real data, given that it is possible to model everything on macroscopic data.

For microscopic data acquisition, we have already started creating a few setups based on the model in [113]. The current version would be for multi-view reconstruction (MVS) and insect acquisition/reconstruction. However, once it has been validated, we plan to modify it so that it can also be used for photometric stereo. With these initial tests, we will be able to look at the case of amber, although the scale is not the same, and the cameras and calibrations shall need revision.

Bibliography

[1] R. P. Agarwal, K. Perera, and D. O'Regan. Multiple positive solutions of singular discrete p-Laplacian problems via variational methods. *Advances in Difference Equations*, 2005:1–7, 2005. 40

[2] P. Agrafiotis, K. Karantzalos, A. Georgopoulos, and D. Skarlatos. Correcting image refraction: Towards accurate aerial image-based bathymetry mapping in shallow waters. *Remote Sensing*, 12(2):322, 2020. 46

[3] H. Alzayer, K. Zhang, B. Feng, C. A. Metzler, and J.-B. Huang. Seeing the World through Your Eyes. In *Conference on Computer Vision and Pattern Recognition*, pages 4864–4873, 2024. 47

[4] Autodesk Maya. `https://www.autodesk.com/products/maya/`. 29

[5] R. Azzam and N. Bashara. *Ellipsometry and Polarized Light*. Ballard CREOL collection. North-Holland Publishing Company, 1977. 22

[6] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *Pattern Analysis and Machine Intelligence*, 37(8):1670–1687, 2014. 31

[7] A. Barroso-Laguna, S. Munukutla, V. A. Prisacariu, and E. Brachmann. Matching 2D Images in 3D: Metric Relative Pose from Metric Correspondences. In *Conference on Computer Vision and Pattern Recognition*, pages 4852–4863, 2024. 34

[8] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision*, pages 404–417, 2006. 34

[9] M. Ben-Ezra and S. K. Nayar. What does motion reveal about transparency? In *International Conference on Computer Vision*, volume 2, pages 1025–1032, 2003. 47

[10] Blender - A 3D modelling and rendering package. 21, 29

[11] J. F. Blinn. Models of light reflection for computer synthesized pictures. In *Annual Conference on Computer Graphics and Interactive Techniques*, pages 192–198, 1977. 30

[12] O. Bogdan, V. Eckstein, F. Rameau, and J.-C. Bazin. DeepCalib: a deep learning approach for automatic intrinsic calibration of wide field-of-view cameras. In *European Conference on Visual Media Production*, 2018. 34

[13] M. Breuß, E. Cristiani, J.-D. Durou, M. Falcone, and O. Vogel. Perspective Shape from Shading: Ambiguity Analysis and Numerical Approximations. *SIAM Journal on Imaging Sciences*, 5(1):311–342, 2012. 31

[14] B. Brument, R. Bruneau, Y. Quéau, J. Mélou, F. B. Lauze, J.-D. Durou, and L. Calvet. RNb-NeuS: Reflectance and Normal-based Multi-View 3D Reconstruction. In *Conference on Computer Vision and Pattern Recognition*, pages 5230–5239, 2024. 45, 55

[15] B. Brument, L. Calvet, R. Bruneau, J. Mélou, S. Gasparini, Y. Quéau, F. Lauze, and J.-D. Durou. A Shape-from-silhouette Method for 3D-reconstruction of a Convex Polyhedron. In *International Conference on Quality Control by Artificial Vision*, 2023. 13, 49, 51

[16] R. Bruneau, B. Brument, L. Calvet, M. Cassidy, J. Mélou, Y. Quéau, J.-D. Durou, and F. Lauze. Multi-view stereo of an object immersed in a refractive medium. *Journal of Electronic Imaging*, 33(3):033005–033005, 2024. 23, 51

[17] N. D. F. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla. Using Multiple Hypotheses to Improve Depth-Maps for Multi-View Stereo. In *European Conference on Computer Vision*, pages 766–779, 2008. 37

[18] B. Cao, R. Deng, and S. Zhu. Universal algorithm for water depth refraction correction in through-water stereo remote sensing. *International Journal of Applied Earth Observation and Geoinformation*, 91:102108, 2020. 47

[19] X. Cao and T. Taketomi. SuperNormal: Neural Surface Reconstruction via Multi-View Normal Integration. In *Conference on Computer Vision and Pattern Recognition*, pages 20581–20590, 2024. 45, 70

[20] M. Cassidy, J. Mélou, Y. Quéau, F. Lauze, and J.-D. Durou. Refractive Multi-view Stereo. In *International Conference on 3D Vision*, 2020. 51

[21] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al. Shapenet: An information-rich 3D model repository. *arXiv preprint arXiv:1512.03012*, 2015. 40

[22] V. Chari and P. Sturm. Multi-View Geometry of the Refractive Plane. In *British Machine Vision Conference*, pages 1–11, 2009. 47

[23] C. Chen, H. Wang, Z. Zhang, and F. Gao. Three-dimensional reconstruction from a fringe projection system through a planar transparent medium. *Optics Express*, 30(19):34824–34834, 2022. 46

[24] H. Chen, C. Li, and G. H. Lee. Neusg: Neural implicit surface reconstruction with 3D Gaussian splatting guidance. *arXiv preprint arXiv:2312.00846*, 2023. 45

[25] W. Chen, H. Ling, J. Gao, E. Smith, J. Lehtinen, A. Jacobson, and S. Fidler. Learning to Predict 3D Objects with an Interpolation-based Differentiable Renderer. *Advances in Neural Information Processing Systems*, 32, 2019. 40

[26] Z. Chen, K.-Y. K. Wong, Y. Matsushita, and X. Zhu. Depth from refraction using a transparent medium with unknown pose and refractive index. *International Journal of Computer Vision*, 102(1–3):3–17, 2013. 47

[27] Colmap. `https://colmap.github.io/`. 40

[28] D. DeTone, T. Malisiewicz, and A. Rabinovich. SuperPoint: Self-Supervised Interest Point Detection and Description. In *Conference on Computer Vision and Pattern Recognition*, pages 224–236, 2018. 34

[29] S. Donné and A. Geiger. Learning Non-Volumetric Depth Fusion Using Successive Reprojections. In *Conference on Computer Vision and Pattern Recognition*, pages 7626–7635, 2019. 40

[30] C. Esteban and F. Schmitt. Silhouette and stereo fusion for 3D object modeling. In *3D Digital Imaging and Modeling*, pages 46–53, 2003. 31, 36

[31] H. Fan, L. Qi, C. Chen, Y. Rao, L. Kong, J. Dong, and H. Yu. Underwater optical 3-D reconstruction of photometric stereo considering light refraction and attenuation. *Journal of Oceanic Engineering*, 47(1):46–58, 2021. 46, 53

[32] Y. Furukawa and J. Ponce. Accurate, Dense, and Robust Multiview Stereopsis. *Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2010. 31, 37

[33] C. Gao and N. Ahuja. A Refractive Camera for Acquiring Stereo and Super-resolution Images. In *Conference on Computer Vision and Pattern Recognition*, pages 2316–2323, 2006. 47

[34] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280–2292, 2014. 34

[35] J. Geng. Structured-light 3D surface imaging: a tutorial. *Advances in Optics and Photonics*, 3(2):128–160, June 2011. 31

[36] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman. Unsupervised Training for 3D Morphable Model Regression. In *Conference on Computer Vision and Pattern Recognition*, pages 8377–8386, 2018. 40

[37] K. Genova, F. Cole, A. Sud, A. Sarna, and T. Funkhouser. Local Deep Implicit Functions for 3D Shape. In *Conference on Computer Vision and Pattern Recognition*, pages 4857–4866, 2020. 40

[38] C. Griwodz, S. Gasparini, L. Calvet, P. Gurdjos, F. Castan, B. Maujean, G. Lillo, and Y. Lanthony. AliceVision Meshroom: An open-source 3D reconstruction pipeline. In *ACM Multimedia Systems Conference*, 2021. 23, 36, 37, 40

[39] A. Guédon and V. Lepetit. SuGaR: Surface-Aligned Gaussian Splatting for Efficient 3D Mesh Reconstruction and High-Quality Mesh Rendering. In *Conference on Computer Vision and Pattern Recognition*, 2024. 46, 70

[40] S. Haner and K. Åström. Absolute pose for cameras under flat refractive interfaces. In *Conference on Computer Vision and Pattern Recognition*, pages 1428–1436, 2015. 47

[41] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Second edition, 2004. 35

[42] R. I. Hartley. In defense of the eight-point algorithm. *Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997. 35

[43] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Conference on Computer Vision and Pattern Recognition*, pages 1106–1112, 1997. 34

[44] H. Henderickx, E. Perkovsky, L. Hoorebeke, and M. Boone. The first pseudogarypid in Rovno amber (Ukraine) (Pseudoscorpiones: Pseudogarypidae). *Palaeontologia Electronica*, Jan. 2013. 19

[45] P. Henzler, V. Rasche, T. Ropinski, and T. Ritschel. Single-image Tomography: 3D Volumes from 2D Cranial X-Rays. 37(2):377–388, 2018. 40

[46] B. K. P. Horn. *Shape From Shading: A Method for Obtaining the Shape of a Smooth Opaque Object From One View.* PhD Thesis, MIT, Department of Electrical Engineering and Computer Science, 1970. 31, 37, 39

[47] K. Ichimaru and H. Kawasaki. Underwater Stereo Using Refraction-Free Image Synthesized From Light Field Camera. In *International Conference on Image Processing*, pages 1039–1043, 2019. 46

[48] S. Ikehata. PS-Transformer: Learning Sparse Photometric Stereo Network using Self-Attention Mechanism. *arXiv preprint arXiv:2211.11386*, 2022. 39

[49] C. Jiang, A. Sud, A. Makadia, J. Huang, M. Nießner, T. Funkhouser, et al. Local Implicit Grid Representations for 3D Scenes. In *Conference on Computer Vision and Pattern Recognition*, pages 6001–6010, 2020. 40

[50] H. Jiang, C. Zeng, R. Chen, S. Liang, Y. Han, Y. Gao, and C. Wang. Depth-neus: Neural implicit surfaces learning for multi-view reconstruction based on depth information optimization. *arXiv preprint arXiv:2303.17088*, 2023. 45

[51] N. Jiang, Z. Cui, and P. Tan. A Global Linear Method for Camera Pose Registration. In *International Conference on Computer Vision*, pages 481–488, 2013. 40

[52] A. Jordt, K. Köser, and R. Koch. Refractive 3D reconstruction on underwater images. *Methods in Oceanography*, 15–16:90–113, 2016. 46

[53] A. Jordt-Sedlazeck, D. Jung, and R. Koch. Refractive Plane Sweep for Underwater Images. In *German Conference on Pattern Recognition*, pages 333–342, 2013. 46

[54] A. Kar, C. Häne, and J. Malik. Learning a Multi-View Stereo Machine. *Advances in Neural Information Processing Systems*, 30, 2017. 40

[55] Y. Kasten, A. Geifman, M. Galun, and R. Basri. Algebraic Characterization of Essential Matrices and Their Averaging in Multiview Settings. In *International Conference on Computer Vision*, pages 5895–5903, 2019. 40

[56] B. Kaya, S. Kumar, C. Oliveira, V. Ferrari, and L. Van Gool. Multi-View Photometric Stereo Revisited. In *Winter Conference on Applications of Computer Vision*, pages 3126–3135, 2023. 45

[57] B. Kaya, S. Kumar, F. Sarno, V. Ferrari, and L. Van Gool. Neural radiance fields approach to deep multi-view photometric stereo. In *Winter Conference on Applications of Computer Vision*, pages 1965–1977, 2022. 45

[58] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson Surface Reconstruction. In *Symposium on Geometry Processing*. The Eurographics Association, 2006. 36

[59] A. Kendall, M. Grimes, and R. Cipolla. PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization. In *International Conference on Computer Vision*, pages 2938–2946, 2015. 35

[60] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4), July 2023. 46

[61] D. H. Lee, I.-S. Kweon, and R. Cipolla. A biprism-stereo camera system. In *Conference on Computer Vision and Pattern Recognition*, volume 1, 1999. 47

[62] V. Lepetit, F. Moreno-Noguer, and P. Fua. EPnP: An accurate O(n) solution to the PnP problem. *International Journal of Computer Vision*, 81:155–166, 2009. 35

[63] V. Leroy, J.-S. Franco, and E. Boyer. Shape Reconstruction Using Volume Sweeping and Learned Photoconsistency. In *European Conference on Computer Vision*, Sept. 2018. 40

[64] M. Li, Z. Zhou, Z. Wu, B. Shi, C. Diao, and P. Tan. Multi-View Photometric Stereo: A Robust Solution and Benchmark Dataset for Spatially Varying Isotropic Materials, 2020. 56

[65] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen. Differentiable Monte Carlo Ray Tracing through Edge Sampling. *ACM Transactions on Graphics*, 37(6):1–11, 2018. 40

[66] Z. Li, T. Müller, A. Evans, R. H. Taylor, M. Unberath, M.-Y. Liu, and C.-H. Lin. Neuralangelo: High-Fidelity Neural Surface Reconstruction. In *Conference on Computer Vision and Pattern Recognition*, 2023. 40

[67] Z. Li, Y.-Y. Yeh, and M. Chandraker. Through the Looking Glass: Neural 3D Reconstruction of Transparent Shapes. In *Conference on Computer Vision and Pattern Recognition*, pages 1262–1271, 2020. 47

[68] K. Liao, L. Nie, S. Huang, C. Lin, J. Zhang, Y. Zhao, M. Gabbouj, and D. Tao. Deep Learning for Camera Calibration and Beyond: A Survey. Mar. 2023. 34

[69] S. Liu, W. Chen, T. Li, and H. Li. Soft Rasterizer: Differentiable Rendering for Unsupervised Single-View Mesh Reconstruction. *arXiv preprint arXiv:1901.05567*, 2019. 40

[70] F. Logothetis, R. Mecca, and R. Cipolla. Semi-calibrated near field photometric stereo. In *Conference on Computer Vision and Pattern Recognition*, pages 941–950, 2017. 40

[71] S. Lombardi, T. Simon, J. Saragih, G. Schwartz, A. Lehrmann, and Y. Sheikh. Neural volumes: learning dynamic renderable volumes from images. *ACM Transactions on Graphics*, 38(4):1–14, July 2019. 43

[72] X. Long, C. Lin, P. Wang, T. Komura, and W. Wang. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In *European Conference on Computer Vision*, pages 210–227, 2022. 45

[73] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *Annual Conference on Computer Graphics and Interactive Techniques*, page 163–169, 1987. 43

[74] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 34

[75] T. Łuczyński, M. Pfingsthorn, and A. Birk. Image rectification with the pinax camera model in underwater stereo systems with verged cameras. In *OCEANS*, pages 1–7, 2017. 46

[76] J. Lyu, B. Wu, D. Lischinski, D. Cohen-Or, and H. Huang. Differentiable refraction-tracing for mesh reconstruction of transparent objects. *ACM Transactions on Graphics*, 39(6):1–13, 2020. 47

[77] H. G. Maas. New developments in multimedia photogrammetry. In *Optical 3D Measurement Techniques III*, 1995. 46

[78] N. Max. Optical models for direct volume rendering. *Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. 42

[79] P. Merrell, A. Akbarzadeh, L. Wang, P. Mordohai, J.-M. Frahm, R. Yang, D. Nister, and M. Pollefeys. Real-Time Visibility-Based Fusion of Depth Maps. In *International Conference on Computer Vision*, pages 1–8, 2007. 37

[80] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019. 40

[81] N. Metropolis and S. Ulam. The Monte Carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949. 30

[82] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. *ACM Transactions on Graphics*, 2019. 43

[83] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *European Conference on Computer Vision*, 2020. 25, 40, 41, 42, 43, 44, 52

[84] N. J. W. Morris. Image-based water surface reconstruction with refractive stereo. Master's thesis, Department of Computer Science, University of Toronto, 2004. 46

[85] N. J. W. Morris and K. N. Kutulakos. Dynamic Refraction Stereo. *Pattern Analysis and Machine Intelligence*, 33(8):1518–1531, 2011. 46

[86] P. Moulon. *Positionnement robuste et précis de réseaux d'images.* PhD Thesis, Université Paris-Est, Jan. 2014. 31

[87] T. Müller, A. Evans, C. Schied, and A. Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics*, 41(4):1–15, 2022. 60

[88] S. Murai, M.-Y. J. Kuo, R. Kawahara, S. Nobuhara, and K. Nishino. Surface normals and shape from water. In *International Conference on Computer Vision*, pages 7830–7838, 2019. 46

[89] T. Murase, M. Tanaka, T. Tani, Y. Miyashita, N. Ohkawa, S. Ishiguro, Y. Suzuki, H. Kayanne, and H. Yamano. A Photogrammetric Correction Procedure for Light Refraction Effects at a Two-Medium Boundary. *Photogrammetric Engineering and Remote Sensing*, 9(8):1129–1136, 2008. 47

[90] Z. Murez, T. Treibitz, R. Ramamoorthi, and D. Kriegman. Photometric stereo in a scattering medium. In *International Conference on Computer Vision*, pages 3415–3423, 2015. 46

[91] J. Mélou. *Fusion d'approches photométriques et géométriques pour la création de modèles 3D.* PhD Thesis, Université de Toulouse, 2020. 30, 31

[92] S. G. Narasimhan, S. K. Nayar, B. Sun, and S. J. Koppal. Structured light in scattering media. In *International Conference on Computer Vision*, volume 1, pages 420–427, 2005. 46

[93] S. Nayar and Y. Nakagawa. Shape from focus. *Pattern Analysis and Machine Intelligence*, 16(8):824–831, 1994. 31

[94] T.-N. Nguyen, H.-H. Huynh, and J. Meunier. 3D reconstruction with time-of-flight depth camera and multiple mirrors. *IEEE Access*, 6:38106–38114, 2018. 47

[95] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger. Differentiable Volumetric Rendering: Learning Implicit 3D Representations without 3D Supervision. In *Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 40

[96] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger. Differentiable volumetric rendering: Learning implicit 3D representations without 3D supervision. In *Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 40

[97] Nvidia. `https://www.nvidia.com`. 29

[98] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019. 40

[99] B. T. Phong. Illumination for computer generated pictures. In *Seminal graphics: pioneering efforts that shaped the field*, pages 95–101. 1998. 29

[100] Y. Quéau. *Reconstruction tridimensionnelle par stéréophotométrie*. PhD Thesis, Université de Toulouse, 2015. 31

[101] Y. Quéau, R. Bruneau, J. Mélou, J.-D. Durou, and F. Lauze. On Photometric Stereo in the Presence of a Refractive Interface. In *Scale Space and Variational Methods*, pages 691–703, 2023. 53, 58

[102] Y. Quéau, T. Wu, and D. Cremers. Semi-calibrated near-light photometric stereo. In *Scale Space and Variational Methods in Computer Vision*, pages 656–668, 2017. 40

[103] Y. Quéau, T. Wu, F. Lauze, J.-D. Durou, and D. Cremers. A non-convex variational approach to photometric stereo under inaccurate lighting. In *Conference on Computer Vision and Pattern Recognition*, pages 350–359, 2017. 39

[104] J. Rainford, M. Hofreiter, D. Nicholson, and P. Mayhew. Phylogenetic Distribution of Extant Richness Suggests Metamorphosis Is a Key Innovation Driving Diversification in Insects. *PloS one*, 9:e109085, Oct. 2014. 16

[105] G. Riegler, A. O. Ulusoy, H. Bischof, and A. Geiger. Octnetfusion: Learning depth fusion from data. In *International Conference on 3D Vision*, pages 57–66, 2017. 40

[106] C. Rockwell, N. Kulkarni, L. Jin, J. J. Park, J. Johnson, and D. F. Fouhey. FAR: Flexible, Accurate and Robust 6DoF Relative Camera Pose Estimation. In *Conference on Computer Vision and Pattern Recognition*, pages 19854–19864, 2024. 35

[107] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An efficient alternative to SIFT or SURF. In *International Conference on Computer Vision*, pages 2564–2571, 2011. 34

[108] S. A. Shafer and T. Kanade. Using shadows in finding surface orientations. *Computer Vision, Graphics, and Image Processing*, 22(1):145–176, 1983. 31

[109] M. Shao, C. Xia, D. Duan, and X. Wang. Polarimetric inverse rendering for transparent shapes reconstruction. *Transactions on Multimedia*, 26:7801–7811, 2024. 47

[110] V. Sitzmann, M. Zollhöfer, and G. Wetzstein. Scene Representation Networks: Continuous 3D-Structure-Aware Neural Scene Representations. *Advances in Neural Information Processing Systems*, 32, 2019. 40

[111] V. Sitzmann, M. Zollhöfer, and G. Wetzstein. Scene representation networks: Continuous 3D-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019. 43

[112] N. Snavely, S. M. Seitz, and R. Szeliski. *Photo tourism: exploring photo collections in 3D*. ACM, 2023. 40

[113] B. Ströbel, S. Schmelzle, N. Blüthgen, and M. Heethoff. An automated device for the digitization and 3D modelling of insects, combining extended-depth-of-field and all-side multi-view imaging. *ZooKeys*, 759:1–27, 2018. 70

[114] P. Sturm. Multi-view geometry for general camera models. In *Conference on Computer Vision and Pattern Recognition*, volume 1, pages 206–212, 2005. 47

[115] A. Tewari, J. Thies, B. Mildenhall, P. Srinivasan, E. Tretschk, W. Yifan, C. Lassner, V. Sitzmann, R. Martin-Brualla, S. Lombardi, et al. Advances in neural rendering. In *Computer Graphics Forum*, volume 41, pages 703–735, 2022. 28

[116] E. Tola, C. Strecha, and P. Fua. Efficient Large Scale Multi-View Stereo for Ultra High Resolution Image Sets. *Machine Vision and Applications*, 23, Sept. 2011. 37

[117] J. Tong, S. Muthu, F. A. Maken, C. Nguyen, and H. Li. Seeing Through the Glass: Neural 3D Reconstruction of Object Inside a Transparent Container. In *Conference on Computer Vision and Pattern Recognition*, pages 12555–12564, June 2023. 47, 52, 55, 62, 66, 70

[118] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle Adjustment — A Modern Synthesis. In *Vision Algorithms: Theory and Practice*, pages 298–372, 2000. 35

[119] R. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *Journal on Robotics and Automation*, 3(4):323–344, 1987. 34

[120] C. Tsiotsios, M. E. Angelopoulou, T.-K. Kim, and A. J. Davison. Backscatter compensated photometric stereo with 3 sources. In *Conference on Computer Vision and Pattern Recognition*, pages 2251–2258, 2014. 46

[121] Unreal Engine. `https://www.unrealengine.com/`. 30

[122] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 25, 26, 40, 44, 47, 52, 55, 60

[123] Y. Wang, Q. Han, M. Habermann, K. Daniilidis, C. Theobalt, and L. Liu. NeuS2: Fast Learning of Neural Implicit Surfaces for Multi-view Reconstruction. In *International Conference on Computer Vision*, pages 3295–3306, 2023. 23, 45, 56, 60

[124] Y. Wang, I. Skorokhodov, and P. Wonka. Hf-neus: Improved surface reconstruction using high-frequency details. *Advances in Neural Information Processing Systems*, 35:1966–1978, 2022. 44, 45

[125] A. P. Witkin. Recovering surface shape and orientation from texture. *Artificial Intelligence*, 17(1):17–45, 1981. 31

[126] A. S. Woodget, J. T. Dietrich, and R. T. Wilson. Quantifying below-water fluvial geomorphic change: The implications of refraction correction, water surface elevations, and spatially variable error. *Remote Sensing*, 11(20):2415, 2019. 47

[127] R. J. Woodham. Photometric Method For Determining Surface Orientation From Multiple Images. *Optical Engineering*, 19(1):134–144, 1980. 37

[128] L. Wu, A. Ganesh, B. Shi, Y. Matsushita, Y. Wang, and Y. Ma. Robust Photometric Stereo via Low-Rank Matrix Completion and Recovery. In *Asian Conference on Computer Vision*, pages 703–717, 2011. 39

[129] T. Wu, J. Wang, X. Pan, X. Xu, C. Theobalt, Z. Liu, and D. Lin. Voxurf: Voxel-based efficient and accurate neural surface reconstruction. *arXiv preprint arXiv:2208.12697*, 2022. 45

[130] X. Wu and X. Tang. Accurate binocular stereo underwater measurement method. *International Journal of Advanced Robotic Systems*, 16(5), 2019. 46

[131] J. Xiong and W. Heidrich. In-the-wild single camera 3D reconstruction through moving water surfaces. In *International Conference on Computer Vision*, pages 12558–12567, 2021. 47

[132] A. Yamashita, Y. Shirane, and T. Kaneko. Monocular Underwater Stereo – 3D Measurement Using Difference of Appearance Depending on Optical Paths. In *International Conference on Intelligent Robots and Systems*, pages 3652–3657, 2010. 47

[133] W. Yang, G. Chen, C. Chen, Z. Chen, and K.-Y. K. Wong. Ps-nerf: Neural inverse rendering for multi-view photometric stereo. In *European Conference on Computer Vision*, pages 266–284, 2022. 45

[134] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 767–783, 2018. 40

[135] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua. Lift: Learned invariant feature transform. In *European Conference on Computer Vision*, pages 467–483, 2016. 34

[136] W. Yifan, F. Serena, S. Wu, C. Öztireli, and O. Sorkine-Hornung. Differentiable surface splatting for point-based geometry processing. *ACM Transactions on Graphics*, 38(6):1–14, Nov. 2019. 40

[137] C. Zhang, X. Zhang, D. Tu, and P. Jin. On-site calibration of underwater stereo vision based on light field. *Optics and Lasers in Engineering*, 121:252–260, 2019. 46

[138] Z. Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000. 34

[139] Z.-Y. Zhang and H.-T. Tsui. 3D reconstruction from a single view of an object and its image in a plane mirror. In *International Conference on Pattern Recognition*, volume 2, pages 1174–1176, 1998. 47

[140] D. Zhao, D. Lichy, P.-N. Perrin, J.-M. Frahm, and S. Sengupta. Mvpsnet: Fast generalizable multi-view photometric stereo. In *International Conference on Computer Vision*, pages 12525–12536, 2023. 40, 45

Chapter A

First paper

A Shape-from-silhouette Method for 3D-reconstruction of a Convex Polyhedron

Baptiste Brument[a], Lilian Calvet[a], Robin Bruneau[a,b], Jean Mélou[a], Simone Gasparini[a], Yvain Quéau[c], François Lauze[b], and Jean-Denis Durou[a]

[a]IRIT, UMR CNRS 5505, Toulouse, France
[b]DIKU, Copenhagen, Denmark
[c]Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, Caen, France

ABSTRACT

We present a pipeline to recover precisely the geometry of a convex polyhedral object from multiple views under circular motion. It is based on the extraction of visible polyhedron vertices from silhouette images and matching across a sequence of images. Compared to standard structure-from-motion pipelines, the method is well suited to the 3D-reconstruction of low-textured and non-Lambertian materials. Experiments on synthetic and real datasets show the efficacy of the proposed framework.

1. INTRODUCTION

Many man-made objects have a relatively simple geometry. Among them, convex polyhedra are abundant, especially rectangular blocks. Estimating their 3D-geometry from the optical acquisition is of interest in multiple application domains. For example, nearly all parcels show polyhedric shapes (Figure 1). Automating their logistics requires knowledge of their dimensions to solve the bin packing problem. Another example is the reconstruction of artifacts encased in polyhedral transparent media (Figure 6). Knowledge of the container geometry is of utmost importance to model the distortion of the image artifact caused by refraction, so as to develop adapted 3D-reconstruction pipelines.

To estimate such a geometry, one could rely on special markers such as AruCo or April Tags (Figure 1e). Yet, this can slow down the acquisition process of large datasets and be cumbersome to manipulate. To avoid markers, we use a semi-controlled image acquisition setup: the block/object is positioned on a turntable and viewed by a static camera whose pose relative to the table is unknown. From the collection of views, we estimate the positions of the polyhedron vertices and the camera poses relative to the polyhedron by combining shape-from-silhouette[1] with point correspondences across the views (Figure 1). This allows one to 3D-reconstruct a polyhedron without assuming Lambertian materials.

The steps of our solution are: extract the silhouette of the polyhedron in each view; detect the edges of the polygon formed by each silhouette; extract the polygon vertices which correspond to the vertices of the imaged polyhedron (Figure 1b); robustly match the imaged polyhedron vertices across the views (Figure 1c); robustly 3D-reconstruct the polyhedron vertices along with its *topology* (Figure 1d). Robustness is important, especially when controlling the lighting is not possible, which may lead to some aberrant silhouette extractions. We apply the method to datasets of real turntable images of five convex polyhedra. As an evaluation of our approach, we also use the geometry and poses information to recover from images a 3D-point cloud of an insect encased in a box-shaped block of epoxy resin using the refractive MVS method.

Our main contribution is thus a robust extraction and matching algorithm for images of points whose trajectories should be parallel 3D-circles. This yields an easy-to-use, marker-free pipeline for 3D-reconstruction of convex polyhedra, which only requires a turntable. Code and datasets will be made publicly available.

After reviewing related work in Section 2, we describe our notations in Section 3. The matching of imaged polyhedron vertices is described in Section 4. An evaluation of the method is then provided in Section 5, and Section 6 draws our conclusions.

Sixteenth International Conference on Quality Control by Artificial Vision,
edited by, Igor Jovančević, Jean-José Orteu, Proc. of SPIE Vol. 12749,
1274918 · © 2023 SPIE · 0277-786X · doi: 10.1117/12.3000368

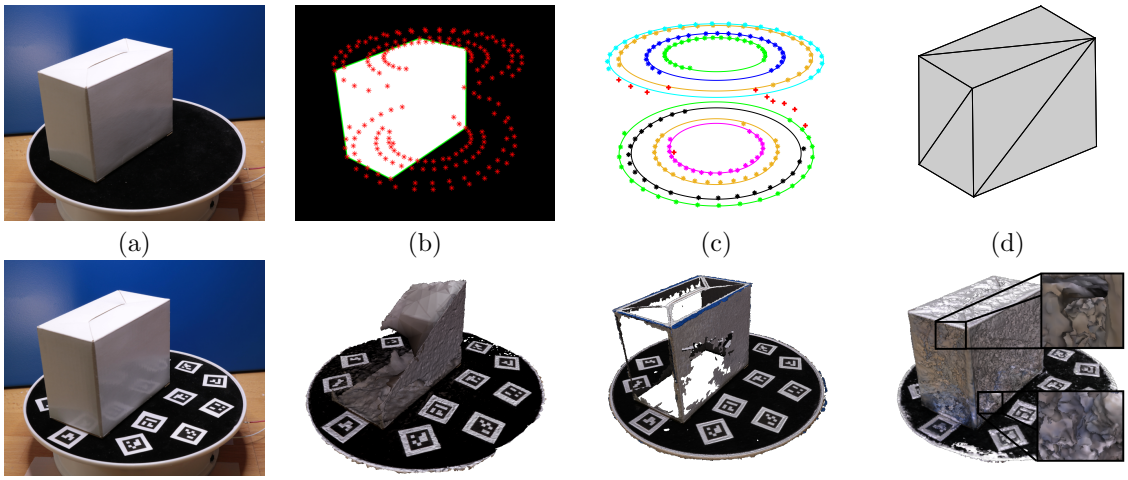


Figure 1. Top row: overview of our 3D-reconstruction pipeline. (a) One (out of 40 images) of a parcel placed on a turntable. The silhouettes vertices, displayed in red in (b), are located on ellipses (c). Our method retrieves the point correspondences from the set of points displayed in (b), namely the subsets of points belonging to the ellipses (c). From these correspondences, we perform the 3D-reconstruction of the polyhedron (d), without using markers. Bottom row: State-the-art results, where the camera poses are computed from markers (e). Multi-view stereo (MVS) results from (f) Meshroom,[2] (g) COLMAP,[3] and (h) the very recent NeRF-based instant-NGP.[4] 3D-reconstruction fails in (f-g) due to the lack of texture and glossy material, while (h) is very noisy.

2. PREVIOUS WORK

Determining the shape of an object placed on a turntable is already a well-studied problem.[5–8] Multi-view geometry can be recovered from epipolar tangents.[8] Yet, this method assumes smooth object, hence it does not work well with polyhedra. Assuming knowledge of the two-view fundamental[5] is also a restrictive assumption. It is for example not possible when reconstructing a tetrahedron comprising four vertices: four point correspondences are available while seven are required to estimate the fundamental matrix, or five in a calibrated scenario.

The geometry can also be recovered by the shape-from-silhouette technique, which computes an object envelope from its silhouettes in different views.[1,9,10] In the case of a polyhedron, reconstructing the planar faces is however impossible in practice because, for every face, the camera optical center must be located in the face supporting plane in at least one view, which is very unlikely. In addition, the camera poses are required and must be computed beforehand. A workaround could consist in placing planar markers[11,12] on the turntable, to compute camera poses before doing the 3D-reconstruction either by shape-from-silhouette or multi-view stereo (MVS). Yet, shape-from-silhouette would fail due to markers moving along with the polyhedron, making the silhouette extraction fail in most views. On the other hand, MVS[13,14] does not perform well in the case of a poorly textured object or non-Lambertian materials (Figure 1f), or for transparent objects (Figure 6e). In the latter case, MVS delivers a mesh comprising a large number of vertices and edges which do not correspond to the real polyhedron faces. Retrieving the latter requires complex post-processing: mesh denoising, holes filling, face segmentation, plane fitting, and plane intersection computation. On the contrary, our method computes the polyhedron faces directly.

The method we propose takes inspiration from the work of Jiang et al.,[6,7] who remark that the trajectories of the points on an object placed on a turntable are parallel 3D-circles whose centres are located on the rotation axis. Their images, therefore, lie on elliptical images of these trajectories. Jiang et al. make use of these ellipses to compute the camera poses. To fit the conics, the authors assume reliable point correspondences across the images collection from marked points. Yet, in our scenario such correspondences are not available and must be computed automatically. This can be particularly challenging in the case of non-Lambertian materials, for which tracking or wide baseline keypoint-based matching algorithms[15–18] perform poorly. Instead, we use the imaged vertices of the object obtained from its polygonal silhouettes.

3. NOTATIONS

We consider a scene consisting of a convex polyhedron with N vertices, placed on a turntable. The vertices coordinates $\mathbf{X}_n$, $n \in \{1, \ldots, N\}$, are expressed in a 3D-frame $\mathcal{R}_{\text{ref}}$ attached to the turntable whose origin is located at the intersection between the supporting plane of the table and the rotation axis, and its two first axes define the supporting plane of the table. The third axis is a normal vector to the table directed upwards.

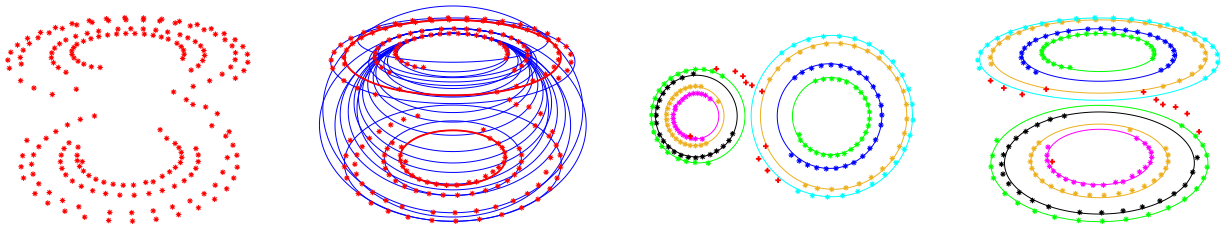
A series of J views of this scene is acquired by a static camera with known intrinsics, and we denote by $\mathbf{x}_n^j$ the image of vertex $\mathbf{X}_n$ in the j^{th} image, $j \in \{1, \ldots, J\}$.

In homogeneous Cartesian coordinates $\mathbf{x} = [x, y, 1]^\top$, an ellipse can be represented as $\mathbf{x}^\top \mathsf{A}\, \mathbf{x} = 0$, where A is a symmetric 3×3 matrix under suitable conditions on its coefficients. The interior (resp. exterior) of the ellipse is given by the points $\mathbf{x}$ for which $\mathbf{x}^\top \mathsf{A}\, \mathbf{x} < 0$ (resp. $\mathbf{x}^\top \mathsf{A}\, \mathbf{x} > 0$).

4. MATCHING OF IMAGED VERTICES

The first step consists in creating polygonal silhouettes of each view. This is obtained by simple operations: background subtraction from a reference image, thresholding, morphological processing, extraction, and simplification of the convex hull. Then the collection V of all the polygon vertices for all the silhouettes is created.

In the second step, V must be robustly partitioned in subsets of points located on common ellipses, as these ellipses should be the images of *parallel circular trajectories* in 3D-space. The partition size, which should match the number of vertices of the polyhedron, is unknown. In the remainder of this section, a *point* refers to an element of V. The cardinality of V is denoted by $|V|$.



(a) (b) (c) (d)

Figure 2. Main steps of the proposed matching. The input data is the silhouettes vertices V in all views, shown in red in (a). *(i)* The ellipses formed by the point triplets showing the highest numbers of points located in their neighborhood are kept, displayed in blue in (b). *(ii)* Robust regression of the imaged circular points from the obtained ellipses. It provides a Euclidean rectification of the points V, displayed in (c), computed from an intersection of a pair of ellipses (in red in (b)) associated with the trajectories of two polyhedron vertices. *(iii)* Collection of the correspondences resulting from the partitioning of points located on circles, displayed in (c), each colour being associated with a vertex. The matching solution is shown in (d). The red crosses are unclassified vertices.

4.1 Exhaustive Search

Five points are needed to determine an ellipse. The partitioning problem can therefore be solved by an exhaustive search for subsets of five points such that a large enough number of the remaining points are located near the ellipse passing through the five selected points. In practice, the number $\binom{|V|}{5}$ of subsets of five points is too high to perform an exhaustive search. Two assumptions enable us to drastically reduce it: we assume that the rotation axis of the turntable projects vertically into the image and that the common abscissa of the centres of the ellipses is known. The estimation of an ellipse is then reduced to the computation of three parameters, namely both semi-axes and the center ordinate, requiring only three points. If the optical axis is roughly pointing towards the rotation axis, then the abscissa of the ellipse formed by the turntable is shared with the ellipses formed by the vertices trajectories. This property makes the abscissa, common to all ellipses centers, easy to get in practice, for example by manually selecting the abscissa of the turntable extremities and considering their mean values. Deviations to our assumptions are evaluated in the supplementary material.[19]

We build the $\binom{|V|}{3}$ triplets of points that can be formed from the point set V. Triplets containing at least two points extracted in the same image are removed since a vertex can only have one image per view. The ellipses are estimated for the set of remaining triplets. The obtained set of ellipses is denoted E. For a polyhedron of general geometry in a noise-free scenario, when the number of views is greater than the number of vertices, i.e. $J > N$, the ellipses of E passing through the largest number of points are the trajectories of the images of the polyhedron vertices, and the associated subsets of points the solution to our matching problem. The computed ellipses are sorted according to the number of points located in their neighborhood. Ellipses with the highest scores should correspond to the subsets of points with the highest probability of representing images of the same vertex. This number of points, or score, is computed as follows.

Score – A point is *located near an ellipse* A if it is inside the elliptical envelope $\mathcal{E}(\mathsf{A}_{+\delta}, \mathsf{A}_{-\delta})$: $\mathsf{A}_{+\delta}$ (resp. $\mathsf{A}_{-\delta}$) denotes the ellipse having the same centre and orientation as A, and whose semi-axes are those of A enlarged (resp. shortened) by δ. This is an approximation of the δ-tubular neighborhood of A proposed in:[20] it avoids the computation of the distance of a point to an ellipse, which involves solving quartic equations and can become demanding. A point with Cartesian homogeneous coordinates $\mathbf{x}$ belongs to $\mathcal{E}(\mathsf{A}_{+\delta}, \mathsf{A}_{-\delta})$ if $(\mathbf{x}^\top \mathsf{A}_{+\delta}\mathbf{x})(\mathbf{x}^\top \mathsf{A}_{-\delta}\mathbf{x}) < 0$, and outside otherwise. The score of an ellipse A is the number of points that are contained in $\mathcal{E}(\mathsf{A}_{+\delta}, \mathsf{A}_{-\delta})$. The value of δ is computed automatically.[19]

Singularity A – The trajectory of a vertex located on the rotation axis of the table is reduced to a point. Such a configuration is a singularity for the score computation. Elliptical envelopes containing this point generally obtain very high scores while associating the images of the vertex with those of other vertices. We circumvent this problem by partitioning the points of V beforehand. Groups consisting of at least four very close points, i.e., located at a distance of less than a given threshold (0.01 in practice) from each other*, are excluded from the score computation.

4.2 Error Pruning

Highest score ellipses may include in some cases images of several vertices, because of the δ-tolerance in the score computation, the presence of measurement noise, or imaged trajectories crossing each other. An example of ellipses associated with the highest scores corresponding to aberrant groupings is shown in Figure 2a. However, ellipses representing the imaged vertices trajectories are among the ellipses of highest scores, reducing consequently the search space.

At this stage, the parallelism of the supporting plane of the vertices trajectories has not been used. This property can be exploited by using a pair of points at infinity, which are conjugate complex points included in the trajectories of all polyhedron vertices. They are the *circular points*[21] of the supporting plane of the turntable. Since the intersections of conics are invariant to any projective transformation,[21] the set of ellipses that are imaged vertices trajectories intersect the images of common conjugate complex points, referred to as ICP (for *images of circular points*). If the ICP are identified, it then becomes possible to classify the set of ellipses collected in Section 4.1 into two classes: those which belong to the family of imaged vertices trajectories, namely when intersecting the ICP, and the others.

ICP identification – In theory, the only set of ellipses in E, whose cardinality is strictly greater than two, that intersect in the same pair of complex conjugate points (the ICP) are the imaged vertices trajectories. This property allows us to compute the ICP of the supporting plane of the table as the solution of a robust regression problem of the intersection points of the ellipses E. Note that chance can produce a subset of at least three ellipses of E intersecting at the same pair of complex conjugate points other than the ICP. It is however highly unlikely that a subset of ellipses intersecting at the same pair of points other than the ICP has a number of elements greater than that of a family of imaged vertices trajectories. Only the subset of ellipses of the highest cardinality, sharing common intersections, is retained.

*After normalizing all the points so that they are located within $[-1, 1]^2$.

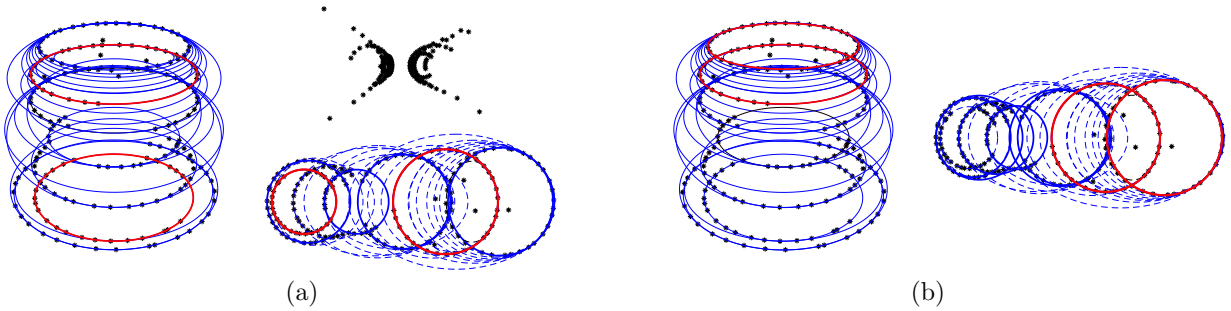
The solution of the robust regression problem is computed as follows. For each pair of ellipses $(\mathsf{A}_1, \mathsf{A}_2) \in E^2$, we count the number of ellipses of $E \setminus \{\mathsf{A}_1, \mathsf{A}_2\}$ passing through the intersections of $(\mathsf{A}_1, \mathsf{A}_2)$. The intersections contained by the largest number of ellipses of E are assumed to be the ICP of the supporting plane of the table.

The proposed solution for the identification of the ICP raises however the following two sub-problems.

Sub-problem 1 – The first sub-problem is how to count the number of ellipses of E passing through the intersections of a pair of ellipses. Two ellipses images of parallel 3D-circles intersect in a pair of real points and a pair of complex points or in two pairs of complex points, one of which is the ICP. In the presence of noise, the ellipses images of the vertices trajectories do not contain the ICP of the table supporting plane. More generally, these ellipses do not intersect in the same pair of points. For this reason, none of the ellipses pass through the ICP in practice. The complex projective space $\mathbb{CP}^2$ is not endowed with a metric that would allow us to use a threshold on the distance from a point to an ellipse, to determine whether or not an ellipse contains a given pair of complex conjugate points. We propose to circumvent this problem by considering that an ellipse A *intersects* the ICP if the ellipse $\mathsf{H}^\top \mathsf{A}\, \mathsf{H}$, rectifying A by some homography

$$\mathsf{H} = [\,\mathbf{h}_1 \;\; \mathbf{h}_2 \;\; *\,], \tag{1}$$

where $\mathbf{h}_1$ and $\mathbf{h}_2$ are the real and imaginary parts of one of the IPC, is a circle. Concretely, the rectification is considered to be a circle if the ratio of the semi-axes of $\mathsf{H}^\top \mathsf{A}\, \mathsf{H}$ is close to 1. For a pair of disjoint ellipses, two homographies composed of the real and imaginary parts of the two pairs of points are applied to the set of ellipses. The homography with the highest number of good rectifications is retained. In the case of a pair of ellipses with real intersections, only the complex conjugate intersections are tested (see Figure 3).



(a) (b)

Figure 3. Example of Euclidean rectifications of a set of points V (black asterisks) from a pair of ellipses, highlighted in red, out of all the K_{best} ellipses (in blue) from the exhaustive search. Ellipses rectified in circles are shown in solid line, in dashed line otherwise. Red circles are the rectifications of the pair of ellipses used to compute the rectifications. (a) The two red ellipses (left) do not intersect, hence two rectifications are therefore computed from their intersections, namely from the two pairs of conjugate complex points. Top: no circle is found. Bottom: four ellipses, highlighted in blue, are rectified into circles. (b) When the red ellipses intersect into two real points, only one rectification is computed. Five ellipses are then rectified into circles. The ellipses rectified into circles are images of polyhedron trajectories.

Sub-problem 2 – The second sub-problem is an excessive cost for computing the intersections of $|E|(|E|-1)/2$ pairs of ellipses of E. To address it, we argued in Subsection 4.2 that the ellipses of the family of imaged vertices trajectories are among the ellipses presenting the best scores. The number of intersection computations is thus reduced to $K_{\text{best}}(K_{\text{best}}-1)/2$, by only considering the K_{best} best ellipses. The value of K_{best} is important but not critical. Its empirically fixed value must be large enough to guarantee that the K_{best} best ellipses include the ellipses images of the trajectories of at least two distinct vertices. We choose $K_{\text{best}} = 5|V|$, which reduces the computation of the intersections to the solution of around $25|V|^2/2$ systems of two quadratic equations in two variables. A result of the ICP computation is shown in Figure 2b, where the pair of ellipses maximising the number of ellipses rectified into circles is shown in red in the rectified representation.

Partitioning – Instead of directly classifying the ellipses intersecting the ICP as belonging to the family of the imaged vertices trajectories and collecting the associated correspondences, Euclidean rectification to the points based on the knowledge of the ICP is first applied and the rectified points are partitioned into subsets of points located on circles. This strategy shows the best results. This can most likely be explained by a distribution of points in the rectified representation that is very well suited to partitioning in cases of ambiguous configurations, such as points distributed along portions of very close ellipses, as seen in the example at the top of Figure 2d, or very low angle shots.

Concretely, the points are rectified by the homography H^{-1} in its form (1). The circles formed by every point triplets of V in the rectified representation are computed. The circles are then sorted in descending order of the score described in Section 4.1. For each circle, processed by descending order of score, the subset of points in its vicinity is removed from V, until the number of points remaining is less than 5. The correspondence solution to our problem is provided by the obtained non-empty subsets of points.

Singularity B – The polyhedron might be positioned such that the images of the trajectories of two vertices lie on the same ellipse. The proposed method cannot separate the two subsets of correspondences in that case. In practice, the user can avoid such a configuration by off-centering the polyhedron with respect to the rotation axis of the table.

The 3D-reconstruction of the polyhedron and camera poses from the ICP $\mathbf{h}_1 \pm i\mathbf{h}_2$ and the imaged vertices correspondences delivered by our matching algorithm can then be performed, following the work of Jiang et al.[7] However, the provided set of correspondences may still comprise some outliers, namely silhouette vertices located on ellipses that are found and which are not imaged polyhedron vertices. The method of Jiang et al. does not explicitly address this problem, hence we modified it to make it robust. The details of the complete method can be found in the supplementary material.[19]

5. EXPERIMENTAL RESULTS

We conducted a large number of experiments with synthetic and real images to quantify the performance of the proposed method.

5.1 Synthetic Data

Synthetic polyhedra are generated as follows. Their vertices are obtained by computing the convex envelope of a set of 3D-points of arbitrary cardinality much greater than eight. The coordinates of the 3D-points are sampled randomly within $[-1, 1]^3$. Points are removed from the convex envelope until sets of 8, 7, 6, 5, and 4 points are obtained, which provides us with five sets of convex polyhedron vertices. The remaining points are scaled so that the three coordinates of all the points $\mathbf{X}_n$ belong to $[-1, 1]$. Forty polyhedra per number of vertices are generated which represents a total of $40 \times 5 = 200$ polyhedra.

The camera resolution is 1000×1000. The focal length is 2000 pixels and the principal point is set at the image center. The camera is located at a randomly selected distance from the centre of rotation of the turntable in the range $[6, 7]$. The camera points towards the table rotation axis. For each polyhedron geometry, five camera tilts are generated, randomly taken nearby a regular angle distribution between -15 and -50 degrees. A series of $J = 35$ angles θ^j of the turntable are randomly sampled according to a Gaussian distribution with 0 mean and a standard deviation of 3 degrees applied on angles regularly distributed between 0 and 360 degrees. Vertices reprojections are generated and, for each view, only those located on the convex hull of all the reprojections are considered as visible. White Gaussian noise of standard deviation varying from 0 to 2 pixels is added to both coordinates of the obtained reprojections. Outliers are introduced in the reprojections bounding box.

The results are expressed using the rate of *successful* matching and the root mean squared error (RMSE) between the estimated vertex positions and the ground truth ones. Given a polyhedron, matching is considered as *successful* if *(i)* the number of partitions found in the matching is equal to the number of ground truth partitions, namely the number of vertices, if *(ii)* there is no misclassified point, and *(iii)* at least three points per partition are found. Matching accuracy, namely the rate of correctly classified matches is also evaluated.

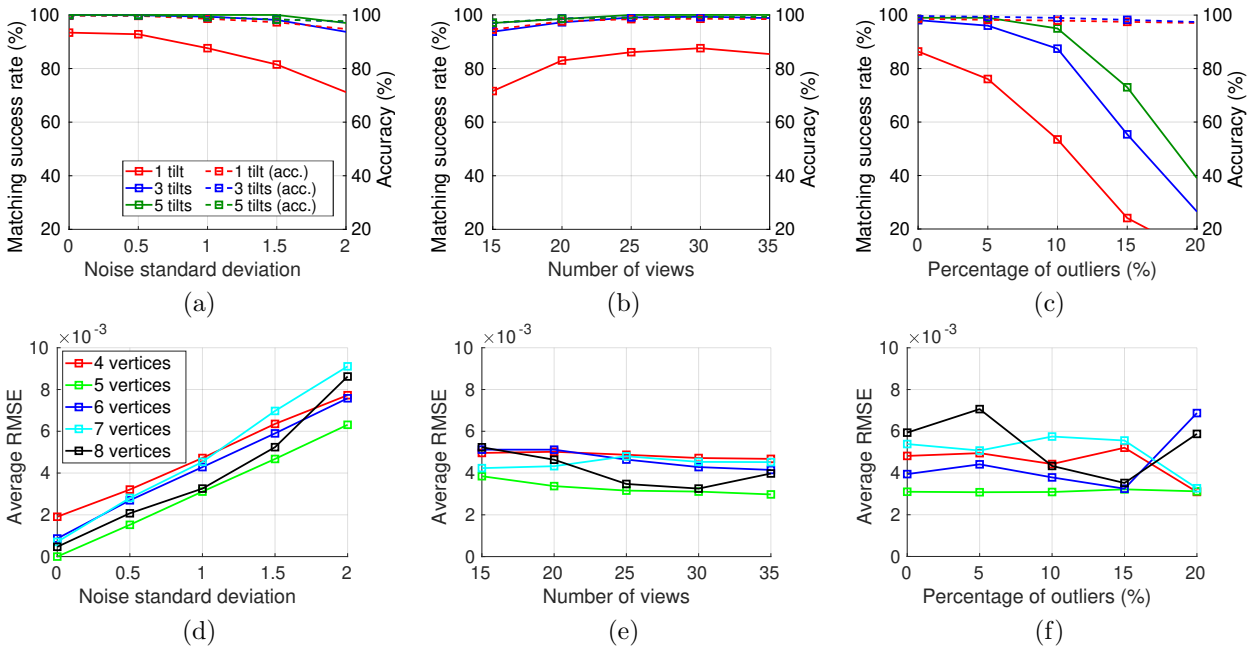


Figure 4. (a,b,c) Success rates (solid lines) and accuracy (dashed lines) of the proposed matching. (d,e,f) RMSE evaluated on the 3D-reconstruction performed when the matching is successful. RMSE reported in (f) has a low correlation with the percentage of outliers, as long as the robustness of the method is not compromised.

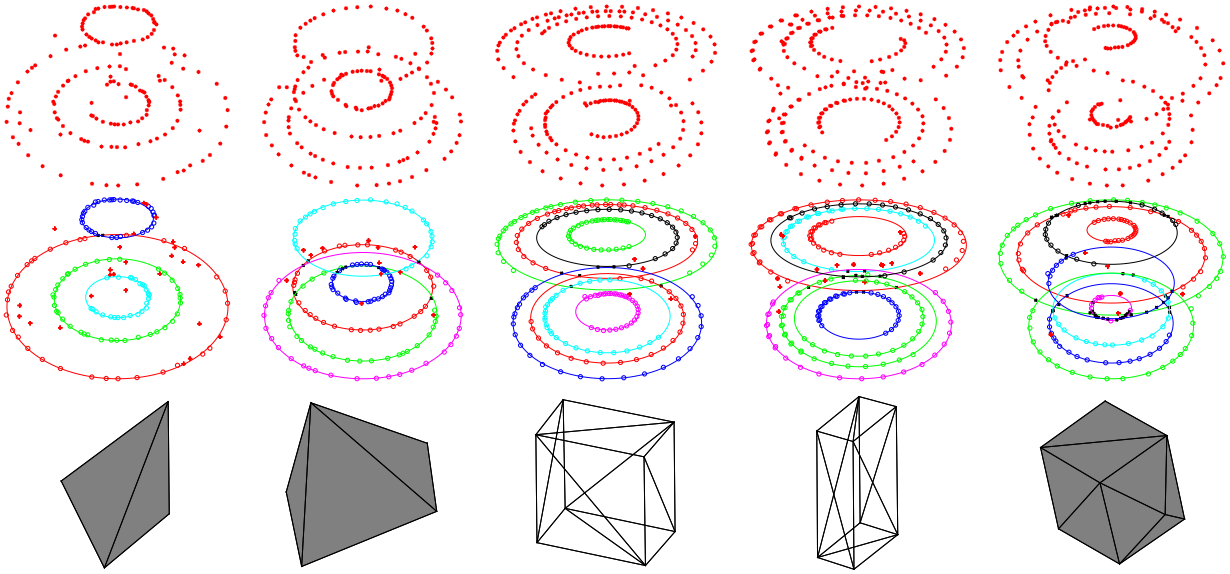
The rates of successful matching and matching accuracy are reported in Figure 5.1a-c, and RMSE evaluated over the successful matching cases in Figure 5.1d-f. They are functions of the level of noise and the number of views, evaluated with 3% of outliers, and function of the amount of outliers.

For each polyhedron, the method is run over all the camera tilts and the best solution is automatically selected. It is the one showing the lowest reprojection error. Acquiring a collection of views of the polyhedron with several camera tilts is straightforward in practice, for example using a standard camera tripod.

The matching shows overall a success rate greater than about 80% apart from the case with lowest number of views (15) and the highest level of noise (2 pixels) and reaches 100% for all polyhedra, from 25 to 35 views and from 0 to 1.5 pixels of noise with 5 camera tilts. Additional results evaluating the proposed ICP estimation method are provided in the supplementary material.[19]

5.2 Real Data

The entire pipeline was evaluated on five real datasets. The first dataset corresponds to the views of a tetrahedron which is the polyhedron with the minimum number of vertices, namely $N = 4$. The other datasets correspond to a square-based pyramid ($N = 5$), two parallelepipeds ($N = 8$), and a 9-vertex polyhedron. The two parallelepipeds are two transparent resin blocks, the first with an encased beetle, the second with an encased grasshopper. The other polyhedra are opaque. The view acquisitions are performed with a single camera tilt. The results are shown in Figure 5. The results of refractive MVS[22] from camera poses and polyhedron computed with markers[12] glued on the polyhedron and with our method are shown in Figure 6. It is therefore possible to use the marker-based results as a control reconstruction to assess the quality of the proposed reconstruction pipeline. MVS fails to reconstruct the resin block. The proposed method shows nearly identical refractive MVS results to the marker-based ones. Results of MVS without refraction[2] for the beetle, results for the grasshopper, and additional results showing that the method performs well under uncontrolled challenging lighting conditions are reported in the supplementary material.[19]



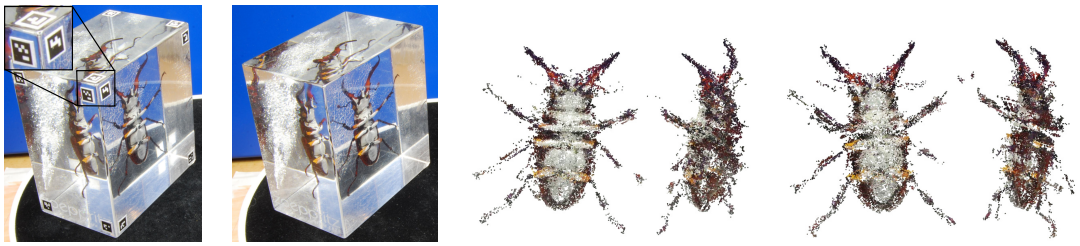
(a) (b) (c) (d) (e)

Figure 5. Top row: Set of points V input of the processing chain for the datasets ***tetrahedron***, ***square pyramid***, ***parallelepiped A***, ***parallelepiped B*** and ***9-vertex polyhedron***. Middle row: Results of the proposed matching. A colour is associated with each vertex. Red crosses indicate unclassified points and black ones indicate ambiguous correspondence class. Both are excluded from the input data for 3D-reconstruction. Bottom row: Solutions of the 3D-reconstructions. The matching and reconstruction succeeds on the five datasets.

6. CONCLUSION

We presented a pipeline for 3D-reconstruction of a convex polyhedron. The method is well suitable for poorly textured and non-Lambertian materials without controlled lighting. Robustness and accuracy of the method is evaluated on synthetic and real data. The method shows to perform well even for transparent medium.

We observed that the matching success drops for a number of vertices greater than 10. We plan to improve our method by adding a final guided matching step so as to retrieve correspondences that are missed using the estimated camera poses.



(a) (b) (c) (d)

Figure 6. (a-b) Excerpt of the input images of a resin block with an encased beetle. (a) Markers are placed on the block to compute a control reconstruction using Meshroom[2] with AprilTag.[12] (b) Block without markers used to evaluate our method. (c) Results of the 3D-reconstructions using refractive MVS[22] from camera poses and interface geometry computed with (c) markers, and (d) our method. The proposed method shows a reconstruction nearly identical to the one computed with the interface geometry obtained using markers. Standard MVS (Meshroom) fails to reconstruct the beetle with camera poses computed with markers (see supplementary material[19]).

REFERENCES

[1] Hernández, C. E. and Schmitt, F., "Silhouette and Stereo Fusion for 3D Object Modeling," *Computer Vision and Image Understanding* **96**(3), 367–392 (2004).

[2] Griwodz, C., Gasparini, S., Calvet, L., Gurdjos, P., Castan, F., Maujean, B., de Lillo, G., and Lanthony, Y., "Alicevision Meshroom: An open-source 3D reconstruction pipeline," in [*Proceedings of MMSys*], (2021).

[3] Schönberger, J. L. and Frahm, J.-M., "Structure-from-motion revisited," in [*Proceedings of CVPR*], (2016).

[4] Müller, T., Evans, A., Schied, C., and Keller, A., "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics* **41**(4) (2022).

[5] Fitzgibbon, A. W., Cross, G., and Zisserman, A., "Automatic 3D Model Construction for Turn-Table Sequences," in [*European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*], (1998).

[6] Jiang, G., Tsui, H.-T., Quan, L., and Zisserman, A., "Single Axis Geometry by Fitting Conics," in [*Proceedings of ECCV*], (2002).

[7] Jiang, G., Tsui, H.-T., Quan, L., and Zisserman, A., "Geometry of Single Axis Motions Using Conic Fitting," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **25**(10), 1343–1348 (2003).

[8] Zhang, H. and Wong, K. K., "Self-calibration of turntable sequences from silhouettes," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **31**(1), 5–14 (2009).

[9] Cheung, G. K. M., Baker, S., and Kanade, T., "Shape-from-silhouette across time part I: theory and algorithms," *International Journal of Computer Vision* **62**(3) (2004).

[10] Cheung, G. K. M., Baker, S., and Kanade, T., "Shape-from-silhouette across time part II: applications to human modeling and markerless motion tracking," *International Journal of Computer Vision* **63**(3) (2005).

[11] Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F. J., and Marín-Jiménez, M. J., "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition* **47**(6), 2280–2292 (2014).

[12] Krogius, M., Haggenmiller, A., and Olson, E., "Flexible layouts for fiducial tags," in [*Proceedings of IROS*], (2019).

[13] Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R., "Nerf: Representing scenes as neural radiance fields for view synthesis," in [*Proceedings of ECCV*], (2020).

[14] Schönberger, J. L., Zheng, E., Pollefeys, M., and Frahm, J.-M., "Pixelwise view selection for unstructured multi-view stereo," in [*Proceedings of ECCV*], (2016).

[15] DeTone, D., Malisiewicz, T., and Rabinovich, A., "Superpoint: Self-supervised interest point detection and description," in [*Proceedings of CVPR*], (2018).

[16] Lowe, D. G., "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision* **60**(2), 91–110 (2004).

[17] Ono, Y., Trulls, E., Fua, P., and Yi, K. M., "Lf-net: Learning local features from images," in [*Proceedings of NeurIPS*], Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., eds. (2018).

[18] Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. R., "ORB: An efficient alternative to SIFT or SURF," in [*Proceedings of ICCV*], (2011).

[19] Brument, B., Calvet, L., Bruneau, R., Mélou, J., Gasparini, S., Quéau, Y., Lauze, F., and Durou, J.-D., "A Shape-from-silhouette Method for 3D-reconstruction of a Convex Polyhedron – Supplementary material."

[20] Kanatani, K. and Ohta, N., "Automatic Detection Of Circular Objects By Ellipse Growing," *International Journal of Image and Graphics* **4**(1), 35–50 (2004).

[21] Hartley, R. and Zisserman, A., [*Multiple View Geometry in Computer Vision*], Cambridge University Press, Second ed. (2004).

[22] Cassidy, M., Mélou, J., Quéau, Y., Lauze, F., and Durou, J.-D., "Refractive multi-view stereo," in [*Proceedings of 3DV*], (2020).

Chapter B

Second paper

JEI Journal of Electronic Imaging

RESEARCH PAPER

Multi-view stereo of an object immersed in a refractive medium

Robin Bruneau[a,b,*] Baptiste Brument[a] Lilian Calvet[c] Matthew Cassidy[a] Jean Mélou[a] Yvain Quéau[d] Jean-Denis Durou[a] and François Lauze[b]

[a]Université de Toulouse, IRIT, UMR CNRS 5505, Toulouse, France
[b]DIKU, Department of Computer Science, Copenhagen, Denmark
[c]University of Zurich, OR-X, Balgrist Hospital, Zurich, Switzerland
[d]Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, Caen, France

ABSTRACT. In this article, we show how to extend the multi-view stereo technique when the object to be reconstructed is inside a transparent but refractive medium, which causes distortions in the images. We provide a theoretical formulation of the problem accounting for a general non-planar shape of the refractive interface, and then a discrete solving method. We also present a pipeline to recover precisely the geometry of the refractive interface, considered as a convex polyhedral object. It is based on the extraction of visible polyhedron vertices from silhouette images and matching across a sequence of images acquired under circular camera motion. These contributions are validated by tests on synthetic and real data.

 [DOI: 10.1117/1.JEI.33.3.033005]

Keywords: three-dimensional reconstruction; multi-view stereo; refraction

Paper 231390G received Dec. 20, 2023; revised Apr. 11, 2024; accepted Apr. 12, 2024; published May 2, 2024.

1 Introduction

Natural history museums often house valuable specimens in transparent mediums, like insects in amber or animals in formaldehyde (see Fig. 1). These specimens are crucial for evolutionary studies but challenging to digitize due to the need for 3D see-through techniques. CT scans are a standard method but are costly and not feasible for large collections. Photogrammetric 3D scanning presents a viable alternative, though it faces challenges due to refraction effects and the shape of the interface between air and the medium. We propose a 3D reconstruction method for objects in a homogeneous refractive medium. Building on previous works,[1,2] this paper contributes a revised algorithm for calculating the shortest optical path between a 3D point and its projection in the target image, for interfaces of any shape. Its main contribution, however, is a comprehensive 3D reconstruction pipeline for objects in refractive media.

1.1 Assumptions

This paper introduces a novel 3D reconstruction method for objects within a refractive medium under the following assumptions: we assume homogeneous refractive media and smooth interfaces, a given index of refraction (or a range), binary masks of the object and of the interface, known camera parameters and triangular mesh representing the interface. In practice, assuming multiple views at fixed rotations on a turntable and visible edges of the medium, camera extrinsics and a convex polyhedron representing the interface can automatically be recovered.

*Address all correspondence to Robin Bruneau, rb@di.ku.dk

1017-9909/2024/$28.00

Fig. 1 (a) Prehistoric beetle trapped in amber (seen under a microscope). (b) Reptiles specimens in jars. Images: A. Solodovnikov (a) and A. D. Jordan (b), courtesy of the Natural History Museum of Denmark.

1.2 Paper Organization

We start by reviewing existing studies on refraction in Sec. 2. Section 3 details the adaptation of the multi-view stereo (MVS) technique in the presence of an interface, focusing on predicting the image projection of a 3D point in the refractive medium, a computationally difficult part. Synthetic image tests in Sec. 4 validate this method. Validation with real data, shown in Sec. 5, involves developing a robust 3D reconstruction method for polyhedral interfaces and an innovative technique for estimating index of refraction without specialized equipment. The paper concludes in Sec. 6, suggesting further extensions.

2 Related Work

Refraction in computer vision varies in treatment: as a bias to correct in classic vision techniques, an element in active systems, or a feature in refractive 3D reconstruction pipelines.

2.1 Refraction Compensation in Classic Vision

Lenses converge light rays from point sources at an image point, essential in optical instruments. Precise lens alignment minimizes aberrations or undesired refraction effects. Transparent objects in a scene can distort the appearance of opaque objects behind them. Studies have addressed these distortions, particularly with transparent objects like window panes attached to cameras, allowing calibration for standard 3D reconstruction pipelines. Maas in Ref. 3 showed how refraction through aquarium glass improves photogrammetry measurements. Łuczyński et al. in Ref. 4 corrected images from underwater cameras to restore epipolar geometry. Image pre-correction has been explored in Refs. 5 and 6, with neural network-based correction in Ref. 7. Light field cameras for refraction correction are discussed in Refs. 8 and 9.

2.2 Active Refraction Techniques

Studies termed active refraction use refraction for single-view 3D reconstruction, duplicating images using bi-prisms[10,11] or rotating glass plates.[12,13]

2.3 Estimation of a Refractive Interface

Morris utilized refracted patterns on water surfaces,[14] and with Kutulakos, mapped points seen through transparency.[15] Ben-Ezra and Nayar[16] fit surface models to distorted images of known geometries. Neural network advancements for 3D reconstruction of transparent objects are noted in Refs. 17 and 18.

2.4 Bathymetry

Refraction correction is essential in remote-sensing bathymetry, and is exemplified by Murase,[19] Woodget,[20] and Cao.[21]

2.5 Classical Framework with Refraction Adaptation

3D vision systems have adapted to refractive interfaces, covering calibration, camera pose estimation, and techniques like refractive structure-from-motion, refractive MVS, and refractive

photometric stereo. Sturm[22] discussed camera models for structure-from-motion, including refractive axial cameras. Chari and Sturm[23] extended epipolar geometry for planar interfaces. Łuczyński et al.[4] proposed a pinhole/axial camera model with calibration, and Chen et al.[24] studied fringe projection systems. Challenges in underwater camera use and implications are detailed in works by Jordt et al.[25–27] and others. Pose optimization under flat refractive interfaces is discussed in Ref. 28, with validations primarily on underwater images.[29] Scenarios like viewing aerial objects from underwater are covered in Refs. 30 and 31 and air to water transitions in Ref. 32. Underwater photometric stereo extensions are investigated in studies like Refs. 33–36.

2.6 Inverse Rendering and Novel Views

Differentiable rasterisers[37–39] and ray tracing inverse renderers[40–42] are emerging in inverse rendering, alongside NeRF adaptations for refraction.[43,44] NeuS[45] and its updated version[46] combine neural Signed Distance Function (SDF) and radiance fields for 3D reconstructions. A framework for objects in cuboid refractive mediums[47] incorporates ambient lighting and ray tracing with Snell–Descartes and Fresnel laws, yet its results are not available for comparison.

We focus on 3D reconstruction by MVS in refractive media, building upon previous works like patch-based MVS,[48] Kang et al.,[49] and Agrawal,[50] targeting also non-planar interfaces, a gap in current research.

3 From Multi-view Stereo to Refracted Multi-view Stereo

3.1 Multi-view Stereo

MVS aims to maximize photometric coherence across different images in a 3D scene for dense 3D reconstruction, as summarized in Ref. 51. Given $t+1$ images and their camera poses, the image of the first pose is chosen as the reference image. Let $\mathbf{P}$ denote a 3D point visible in all images, $\mathbf{p}=\pi(\mathbf{P})$ its projection in the reference image and $\mathbf{p}_j=\pi_j(\mathbf{P})$, $j\in\{1,\dots,t\}$, and its projections in the t other images, called control images. The Lambertian assumption is written

$$I_j\circ\underbrace{\pi_j\circ\pi_z^{-1}(\mathbf{p})}_{\mathbf{p}_j}=I(\mathbf{p}),\quad j\in\{1,\dots,t\},\tag{1}$$

where I_j and I denote the gray level functions of the j'th control and reference images. The index z in π_z^{-1} is necessary as the point $\mathbf{P}=\pi_z^{-1}(\mathbf{p})$ is defined only if its depth z is known.

The MVS technique consists in searching for the point $\mathbf{P}=\pi_z^{-1}(\mathbf{p})$, conjugate of $\mathbf{p}$, satisfying the system of Eqs. (1), by solving, for instance, the least squares problem

$$\min_{z\in\mathbb{R}}\sum_{j=1}^{t}[I_j\circ\pi_j\circ\pi_z^{-1}(\mathbf{p})-I(\mathbf{p})]^2.\tag{2}$$

In practice, the comparison between the gray levels I_j and I is performed between neighborhoods of $\mathbf{p}_j$ and of $\mathbf{p}$, the use of a robust estimator is recommended (see the overview presented in Ref. 51).

When the medium is homogeneous, the π_z^{-1} transformation from the reference view to the 3D scene consists in inverting the central projection. Denoting by $\mathbf{K}$ the camera's calibration matrix, this transformation is written

$$\pi_z^{-1}(\mathbf{p})=z\,\mathbf{K}^{-1}\begin{bmatrix}\mathbf{p}\\1\end{bmatrix}.\tag{3}$$

The reprojection on the j'th control image is also obtained by central projection, considering the camera pose change as a known rigid transformation between the reference pose and the j'th with rotation matrix $\mathbf{R}_j$ and translation vector $\mathbf{t}_j$. With the projection operator $f([a,b,c]^\top)=[a/c,b/c]^\top$, this second transformation is written

$$\pi_j(\mathbf{P})=f(\mathbf{K}(\mathbf{R}_j\mathbf{P}+\mathbf{t}_j)).\tag{4}$$

Carrying over Eqs. (3) and (4) into Eq. (2), the problem of 3D reconstruction by MVS is rewritten

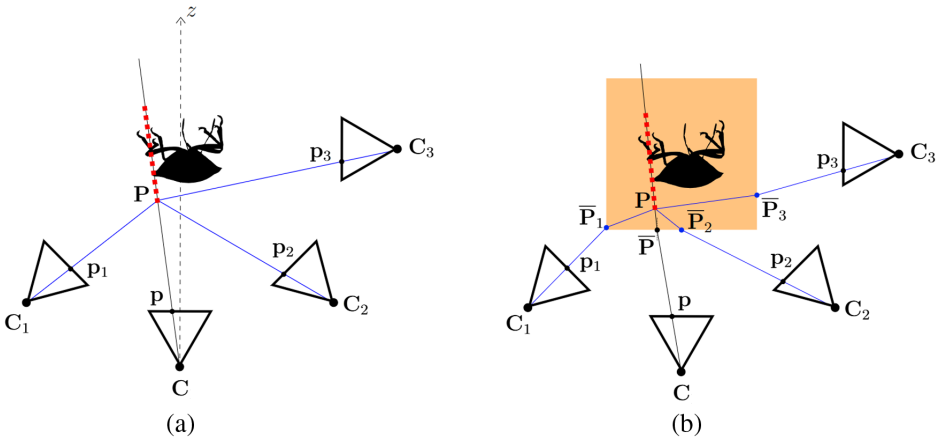


Fig. 2 (a) MVS in a homogeneous medium: the different proposals for the point P, which are materialized by red dots, are reprojected in the control images. (b) MVS with refraction: the reprojection of P in the control images is more difficult to compute, due to refraction.

$$\min_{z\in\mathbb{R}} \sum_{j=1}^{t} [I_j \circ f(\mathbf{K}\,(z\,\mathbf{R}_j\mathbf{K}^{-1}[\begin{smallmatrix}\mathbf{p}\\1\end{smallmatrix}] + \mathbf{t}_j)) - I(\mathbf{p})]^2. \tag{5}$$

The objective in Eq. (5) is nonlinear non-differentiable and/or non-convex, making optimization potentially difficult. Solving Eq. (5) is thus usually done by an exhaustive search (brute-force) in a predefined list of values of depth z [see Fig. 2(a)]. This simplistic strategy has shown to be very effective for the 3D reconstruction of scenes with sufficiently textured surfaces.[52] As Fig. 2(b) indicates, the scenario is more complex when the 3D scene is immersed in a refractive medium.

3.2 Refractive Multi-view Stereo

The image of an object in a refractive medium, with an index of refraction (IoR) over 1, becomes distorted, altering its epipolar geometry. In this context, a point in one image correlates to a curve whose form is influenced by the IoR and the interface shape between the medium and air. Chari and Sturm's work in Ref. 23 generalizes epipolar geometry's matrix formalism with a 12×12 fundamental matrix, important for camera pose estimation in structure-from-motion. Since refraction adaptation in this field is covered in Refs. 27 and 53, our paper focuses on adapting the MVS technique for 3D scenes in refractive mediums. This new challenge, refractive MVS (RMVS), involves addressing Eq. (2) at each point $\mathbf{p}$ in the reference image, with necessary adjustments. In the context of refraction:

- Back-projection of image point $\mathbf{p}$: the back-projection of $\mathbf{p}$ in refractive conditions involves tracing a broken line from $\mathbf{C}$ through $\mathbf{p}$ [see Fig. 2(b)]. The back-projection formula is more complex than Eq. (3), expressed as

$$\pi_{\bar{z}}^{-1}(\mathbf{p}) = \overline{\mathbf{P}} + \bar{z}\mathbf{v}, \tag{6}$$

  where $\overline{\mathbf{P}}$ is the point of incidence at the interface, the unit director vector $\mathbf{v}$ of the refracted ray follows Snell–Descartes refraction law (see Sec. 3.3), and $\bar{z} \geq 0$ is the distance between $\overline{\mathbf{P}}$ and $\mathbf{P}$ along the refracted ray [see Fig. 2(b)]. Determining $\overline{\mathbf{P}}$ varies in complexity with the interface's shape, while computing $\mathbf{v}$ is straightforward if interface normals are accurately known. Tests on synthetic images (with known normals) and real images (assuming a polyhedral interface) are conducted, leaving generalization to any interface shape and effects of normal estimation inaccuracies for future exploration.
- Reprojection of 3D point $\mathbf{P}$: computing the reprojection $\mathbf{p}_j = \pi_j(\mathbf{P})$ with refraction is more complex than Eq. (4), involving solving a shortest optical path problem (see Sec. 3.3).
- Image multiplication: refraction can cause a single 3D point $\mathbf{P}$ to project to multiple image points, as shown in Fig. 6. Each projection is equally viable for solving Eq. (2).

Compared to the MVS technique, the main difficulty of RMVS is the reprojection $\mathbf{p}_j = \pi_j(\mathbf{P})$, $j \in \{1, \ldots, t\}$, of a 3D point $\mathbf{P}$ into the different control images. Let us first consider the case of a planar interface, before tackling the case of an interface of any shape.

3.3 Planar Interface

The first Snell–Descartes law asserts that the refracted ray lies in the plane of incidence, spanned by the incident ray, and the interface normal in $\overline{\mathbf{P}}_j$: the phenomenon is planar [see Fig. 3(a)].

Let i_1 be the angle between the interface normal and the ray in IoR n_1 medium, and i_2 be the angle between the normal and the ray in IoR n_2 medium. The second Snell–Descartes law asserts that

$$n_1 \sin i_1 = n_2 \sin i_2. \tag{7}$$

For a planar interface, squaring both sides of Eq. (7) and using notations from Fig. 3(a), we get

$$n_1^2 \frac{(u_1 - \overline{u})^2}{(u_1 - \overline{u})^2 + v_1^2} = n_2^2 \frac{(u_2 - \overline{u})^2}{(u_2 - \overline{u})^2 + v_2^2}. \tag{8}$$

To find the point of incidence $\overline{\mathbf{P}}_j$ on the u-axis, we need to solve a quartic equation in $\bar{u}$

$$a_4 \overline{u}^4 + a_3 \overline{u}^3 + a_2 \overline{u}^2 + a_1 \overline{u} + a_0 = 0, \tag{9}$$

whose coefficients are based on u_1, v_1, u_2, v_2, and $\alpha = n_2/n_1$.[50] For planar interfaces, Eq. (9) typically has one real solution, found using methods like Newton–Raphson. To compute $\mathbf{p}_j = \pi_j(\mathbf{P})$, first solve Eq. (9) for $\overline{\mathbf{P}}_j$, then project it into the j'th control image as per Eq. (4).

3.4 Interface of Any Shape

The Huygens–Fresnel principle predicts wave surfaces orthogonal to light rays. Dijkstra's algorithm[55] offers a discrete method to calculate these wave surfaces, enabling the shortest path identification between graph vertices. For tracing light rays, the scene can be divided into voxels, serving as the vertices of an undirected graph. The process simplifies in a homogeneous refractive medium, where light propagates straight, similar to air.

The path of a light ray from a 3D point $\mathbf{P}$ to the center of projection $\mathbf{C}_j$ of a control camera, $j \in \{1, \ldots, n\}$, forms a broken line with a single break at the interface, as illustrated in Fig. 2(b). As previously discussed, locating the shortest optical path between $\mathbf{P}$ and $\mathbf{C}_j$ boils down to finding the incidence point $\overline{\mathbf{P}}_j$. Solving this for a planar interface equates to solving a quartic equation (see Sec. 3.3), but it becomes analytically challenging with more complex interface shapes.

One might wonder if the π_j transformation preserves point alignment, specifically if the π_j image of a refracted light ray remains straight in the j'th control image. Figure 3(b) shows that

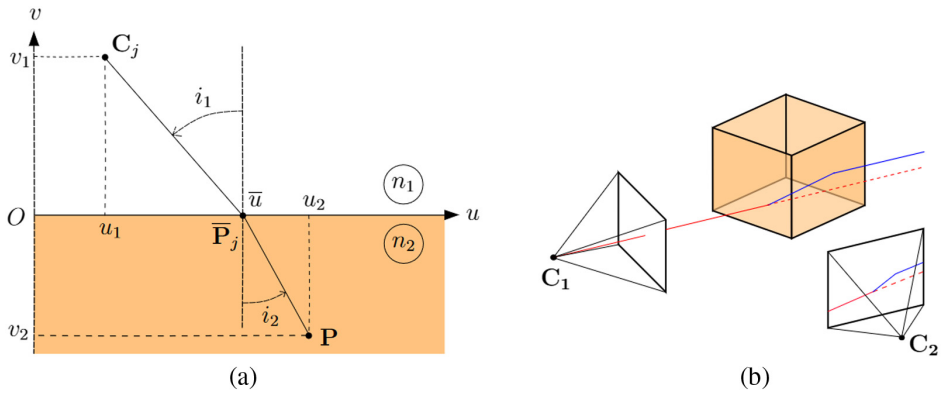


Fig. 3 (a) Second Snell–Descartes law on refraction. (b) The back-projected ray, which has two breaks as it crosses the refractive cube, does not project into the control camera along the red epipolar line. For reasons of clarity, this graphical representation does not perfectly conform to the Snell–Descartes laws.

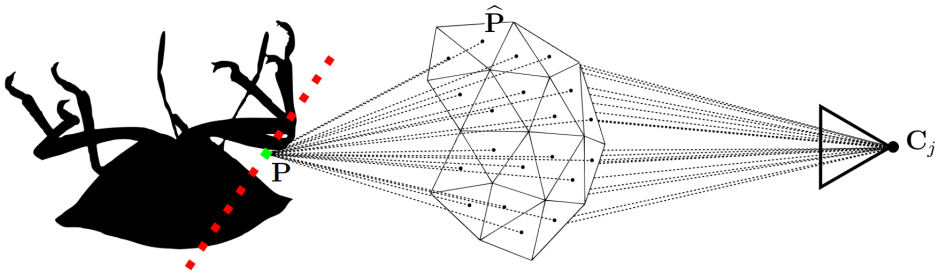


Fig. 4 The point of incidence $\overline{\mathbf{P}}_j$ between a 3D point and the center of projection $\mathbf{C}_j$ of the j'th control camera is determined by testing the set of barycenters $\hat{\mathbf{P}}$ of the triangles of the 3D mesh of the interface that are seen by this camera.

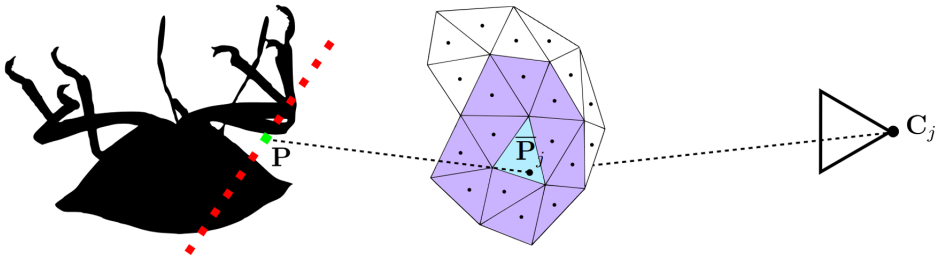


Fig. 5 Once the triangle corresponding to the solution of problem (10) has been identified (triangle indicated in blue), and the search for the point of incidence $\overline{\mathbf{P}}_j$ is refined using the method described in Sec. 3.3. In the case where the solution of this second problem is outside the triangle, a search is performed on the set of adjacent triangles (triangles indicated in purple).

this can be the case with a planar interface, however, with a continuous interface as in Fig. 17, the ray image in no more straight.

For general cases, finding the incidence point $\overline{\mathbf{P}}_j$ involves discretizing the interface and minimizing the optical path of the ray $(\mathbf{C}_j, \hat{\mathbf{P}}, \mathbf{P})$ through potential points $\hat{\mathbf{P}}$ on the discretized interface, via a potentially heavy exhaustive search on "eligible points" $\hat{\mathbf{P}}$

$$\overline{\mathbf{P}}_j = \underset{\hat{\mathbf{P}}}{\operatorname{argmin}}\{n_1 d(\mathbf{C}_j, \hat{\mathbf{P}}) + n_2 d(\hat{\mathbf{P}}, \mathbf{P})\}, \tag{10}$$

where $d(\cdot,\cdot)$ denotes the Euclidean distance in $\mathbb{R}^3$.

Practically, the interface is discretized into a 3D mesh with triangular faces. The eligible points $\hat{\mathbf{P}}$ for incidence point search are the barycenters of the mesh triangles visible from the projection center $\mathbf{C}_j$, as depicted in Fig. 4. The solution of Problem (10) corresponds to the blue-colored triangle in Fig. 5. To refine this result, $\overline{\mathbf{P}}_j$ is then sought in the plane of this triangle, following the method in Sec. 3.3. The solution is accepted if it is inside the triangle. If not, a similar search is conducted on all adjacent triangles (colored purple in Fig. 5). In the absence of a solution within the triangles, the initial solution of Problem (10) is chosen as the incidence point. A more precise search involving optimization under linear constraints defining the triangle is possible but significantly increases computation time. Therefore, despite testing this approach, it has been omitted from our current methodology.

4 Validation on Synthetic Images

4.1 Cubic Interface

We begin by validating our method on a scene featuring a graphosoma insect, approximately 30 *mm* in size, immersed in a refractive cube with an IoR matching that of epoxy resin ($n_2 = 1.56$). The focal length of the camera is 50 *mm*, with an average distance of about 180 *mm* from the scene. Figure 6 displays two synthetic images (out of a total of 18) of this scene, generated using the ray tracing capabilities of Blender software.

Fig. 6 Two synthetic images (among 18) of a graphosoma immersed in a cube of epoxy resin. In both cases, the insect is visible through three faces of the cube (only partially, regarding the top face). Due to reflection phenomena, fragments of the object are visible at the boarders of the immersion medium. These image fragments have not been used by our solving method. Source of the 3D model: LIRIS's datasets library.[54]

Fig. 7 3D reconstruction of a graphosoma immersed in a cube of epoxy resin, seen from three angles, obtained by our RMVS solving method from 18 synthetic images such as those in Fig. 6.

Figure 7 presents three views of the colored 3D point cloud reconstructed using our RMVS method. Figure 8 displays this point cloud post-processing, where it has been "cleaned" using the Connected-component labeling tool in the Cloud Compare software.

Figure 9 compares results obtained without considering refraction, utilizing three different algorithms: a basic MVS from Eq. (5), the Meshroom-proposed MVS pipeline[56] aligned with state-of-the-art algorithms, and neural reconstruction with NeuS2.[46] These methods, not accounting for refraction, fail to interpret the distortions and image duplications of the graphosoma,

Fig. 8 Result of Fig. 7 after "cleaning" the 3D point cloud by the Connected-component labeling tool of the Cloud Compare software.

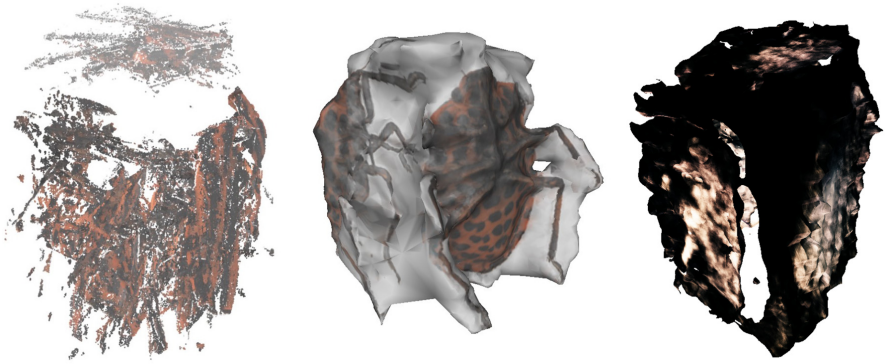


Fig. 9 3D reconstruction results without considering refraction. From left to right: a basic MVS derived from Eq. (5); the MVS approach by Meshroom;[56] the neural 3D surface reconstruction method NeuS2.[46] The duplication of the graphosoma caused by refraction leads to inaccurate reconstructions.

Table 1 Root mean square error (RMSE, in *mm*) comparison between our method and three other methods, which do not take refraction into account. The lowest score (in bold) is unsurprisingly our method which takes refraction into account.

Method	Basic MVS	Meshroom	NeuS2	Ours
RMSE (*mm*)	6.12	5.44	6.44	**0.57**

leading to poorly reconstructed scenes (a comparison with ReNeuS (Ref. 47), a refraction-inclusive extension of NeuS, would be ideal, but its code is unavailable). Table 1 confirms these shortcomings with high root mean square error (RMSE) scores. Conversely, our RMVS solving method, tailored for refraction, demands significantly more computational time: from 5 to 16 min for Fig. 9's results to 24 h for Fig. 7's reconstruction (using CPU Intel Xeon Silver 4110 2.10 GHz with all 32 threads for parallel computing), for roughly 500,000 3D points in each instance. Notably, the computation time has been reduced since only those barycenters $\hat{\mathbf{P}}$ of the mesh triangles (see Fig. 4) that project within the insect's silhouette in all the control images are considered.

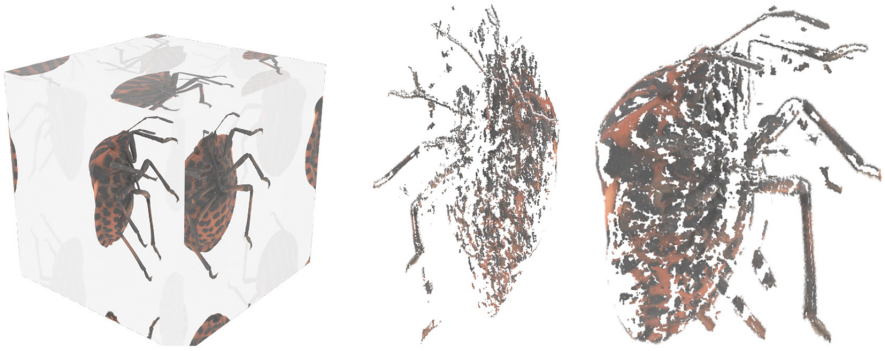
This first example provides insights into our solving method. The 3D reconstruction in Fig. 7 is derived from merging eight colored 3D point clouds, each generated as follows.

- One image is selected as the reference image. Five others serve as control images: in four, the main image of the insect is viewed through the same cube face as in the reference image; in the fifth, it is through an adjacent face.
- For each pixel $\mathbf{p}$ in the reference image, we consider each point $\mathbf{P}$ on the refracted ray from the back-projection of $\mathbf{p}$, for each control image, and for each cube face visible in the control image (up to three per control image). A quartic equation of type Eq. (9) is then solved using the Newton–Raphson method.
- After finding a solution, if its projection $\mathbf{p}_j$ in the j'th control image falls inside the insect's silhouette on the relevant face, the similarity between the neighbourhoods of $\mathbf{p}$ and $\mathbf{p}_j$ is computed using a robust estimator, here sum of absolute deviations (SAD). If the SAD is calculable for multiple faces in the j'th control image (up to three), only the smallest value is kept. If no SAD can be calculated, another point $\mathbf{P}$ is tested from a predefined list of 3D points. The chosen point $\mathbf{P}$ is the one that minimizes the SAD, it is assigned the color of the pixel $\mathbf{p}$ in the reference image. If no SAD can be calculated, no 3D point is associated with pixel $\mathbf{p}$.

Since all the graphosoma images are synthetic, we can measure the deviations from the ground truth for each of the eight 3D point clouds whose fusion yields the result in Fig. 7. Table 2 lists the square root of both the mean and median of these squared deviations.

Table 2 Second line: root mean square error (RMSE, in *mm*) of the eight 3D point clouds whose fusion provides the result of Fig. 7. Third line: root median square error (RMedSE, in *mm*). The last column gives these estimates for all eight 3D point clouds.

Face	Front	Front-right	Right	Back-right	Back	Back-left	Left	Front-left	All
RMSE (*mm*)	0.25	0.48	0.85	0.98	0.40	0.73	0.60	0.65	0.57
RMedSE (*mm*)	0.15	0.30	0.40	0.40	0.25	0.33	0.33	0.40	0.30



(a) (b) (c)

Fig. 10 (a) Example demonstrating the tripling of the graphosoma's image. A 3D point cloud is derived from this single image, selecting the "main" image (right face) as the reference. (b) and (c) Two perspectives of the 3D point cloud reconstructed by our RMVS solving method, using just this single image.

These values are considerably low relative to the scale of the reference 3D model and its distance from the camera, providing quantitative validation for our RMVS solving method.

Figure 10(a) illustrates that the image of a point **P** within a refractive medium can produce multiple images, each representing a local minimum of the optical path between **P** and the projection center (Fermat principle). Thus, it is feasible to match this image of the insect, effectively applying our RMVS solving method with a single view, as demonstrated in Refs. 10 and 11. The result, shown in Figs. 10(b) and 10(c), is rough and incomplete, comprising just a single point cloud. However, this technique differs from other single-view 3D reconstruction methods like shape-from-shading,[57] as it relies on the principle of triangulation.

4.2 Spherical Interface

The second experiment involves a graphosoma immersed in an epoxy resin sphere. Figure 11 presents two synthetic images of this setup, alongside images of the graphosoma from identical angles but outside the refractive medium. The notable differences between these image pairs, apart from the magnification effect of the resin sphere acting like a convex lens are visible in the deformed appearance of the insect's legs and antennae due to refraction. Unlike the images in Fig. 6, the images in Figs. 11(a) and 11(b) are not multiplied. They are rendered using ray tracing, approximating the sphere with a triangular mesh of 327,000 faces, generated in Blender from an icosphere with applied subdivisions.

Figure 12 presents the colored 3D point cloud from 3 angles, reconstructed using our RMVS method from 18 images like those in Figs. 11(a) and 11(b). This cloud is formed by merging eight 3D point clouds. Notably, the legs and antennae of the insect align perfectly across these clouds, and even very fine details are captured. It is important to note that these 3D point clouds are merged with no post-processing, except for cleaning by the Cloud Compare's Connected-component labeling tool. However, the high number of faces on the sphere significantly increases the computation time, from 24 h for the result of Fig. 7 to 1 week for that of Fig. 12.

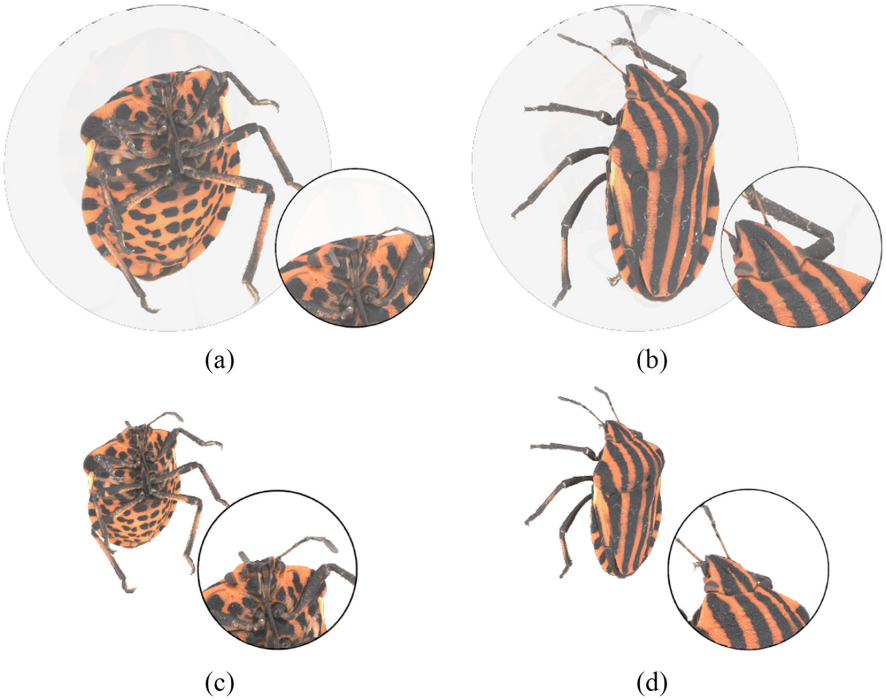


Fig. 11 (a) and (b) Two synthetic images of the graphosoma immersed in an epoxy resin sphere. (c) and (d) Synthetic images of the graphosoma from the same angles but outside the refractive medium. Along with the magnification effect from the resin's convex shape, the insect's legs and antennae appear deformed.

Fig. 12 3D reconstruction of the graphosoma immersed in an epoxy resin sphere, viewed from 3 angles, obtained with our RMVS solving method with 18 images such as those in Figs. 11(a) and 11(b). The reconstruction was refined using the Connected-component labeling tool of the Cloud Compare software.

For comparison, Fig. 13 shows that when refraction is not considered, MVS struggles to accurately reconstruct the 3D shape, resulting in ghosted legs and antennae. This issue highlights the inconsistency among the eight 3D point clouds.

The choice of interface discretization scale balances precision with computing time. Table 3 demonstrates the impact of reducing the number of triangular faces in the sphere's 3D mesh (using Cloud Compare's decimation tool), which implies a less precise interface representation. This is assessed through the same two estimators introduced in Sec. 4.1 (RMSE and RMedSE), along with the percentage of 3D points successfully reconstructed, and the required CPU time.

Figure 14 shows the 3D reconstructions corresponding to Table 3. As anticipated, the first 3D points to "disappear" (those conjugated with pixels p for which no SAD similarity value is calculable) are on the thinnest parts of the 3D model, specifically the legs and antennae.

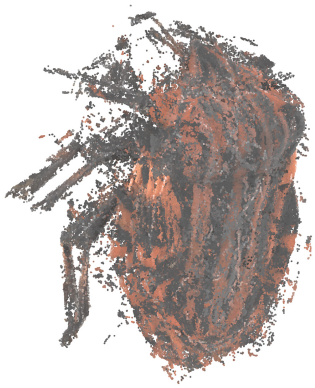


Fig. 13 3D reconstruction using MVS from 18 images such as those in Figs. 11(a) and 11(b) results in ghostly legs and antennae due to inconsistencies among the eight 3D point clouds.

Table 3 Impact of reducing the number of triangular faces in the sphere's 3D mesh on the 3D reconstruction of the graphosoma using our RMVS solving method (the number of faces used to approximate the sphere is indicated in parentheses).

Percentage of faces	100% (327k)	50% (164k)	25% (82k)	10% (33k)	5% (16k)	1% (1.6k)	0.5% (820)
RMSE (*mm*)	1.77	1.82	1.90	2.03	2.25	3.72	4.32
RMedSE (*mm*)	0.30	0.33	0.35	0.45	0.57	2.10	3.05
Reconstructed 3D points	98.0%	97.4%	96.2%	93.9%	91.6%	77.8%	69.4%
CPU time (min)	498	272	147	49	37	28	23

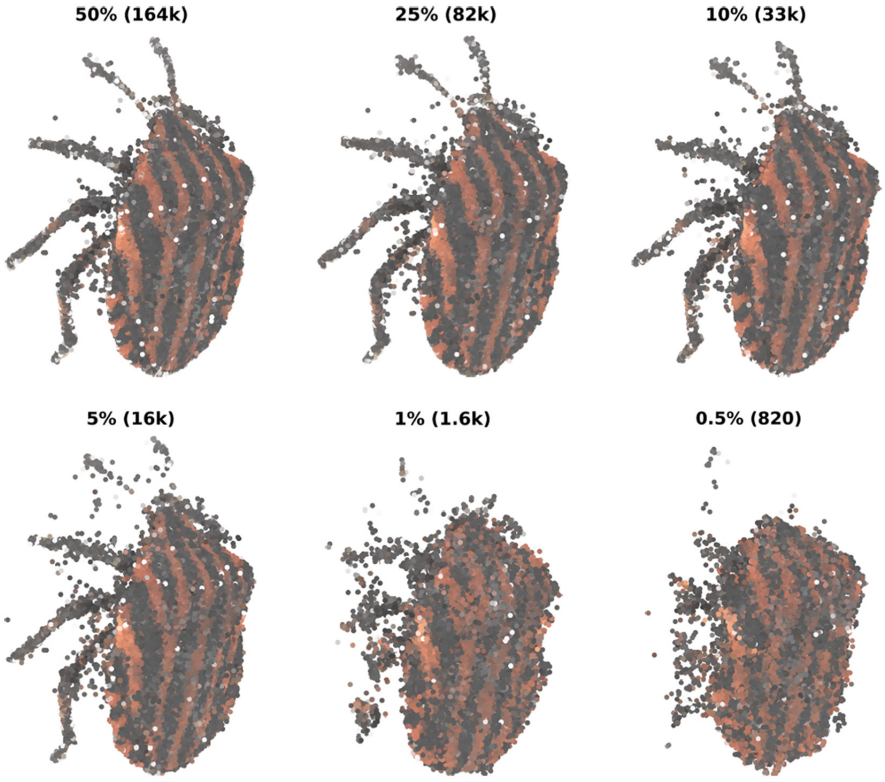


Fig. 14 Six 3D reconstructions of the graphosoma immersed in an epoxy resin sphere, illustrating the effect of reducing the percentage of triangular faces of the sphere used in our RMVS solving method (refer to Table 3). Figure 12 displays the outcome when all 327,000 faces are utilized.

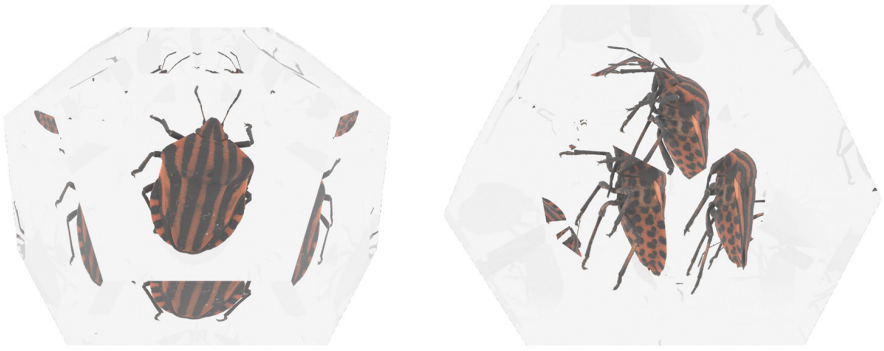


Fig. 15 Two images of the graphosoma immersed in an epoxy resin regular dodecahedron.

Fig. 16 3D reconstruction of the graphosoma immersed in an epoxy resin regular dodecahedron, viewed from three angles. This was created using our RMVS solving method from 18 images such as those in Fig. 15, followed by refinement using the Connected-component labeling tool of the Cloud Compare software.

4.3 Other Interfaces

Figure 15 presents two synthetic images of the graphosoma inside a regular dodecahedron made of the same epoxy resin. Our RMVS 3D reconstruction is shown in Fig. 16. The deviations from the ground truth are only marginally higher than those in Table 2, despite the images being more challenging for 3D interpretation compared to those in Fig. 6. While the RMSE for the entire point cloud increases from 0.57 to 1.10 mm, the RMedSE rises less significantly, from 0.30 to 0.35 mm. This higher RMSE value is likely due to substantial image deformation, potentially skewing the SAD estimator's similarity measurement.

Figure 17 shows that images can undergo even more distortion with a block of any convex shape. The 3D reconstruction achieved by our RMVS solving method, as shown in Fig. 18,

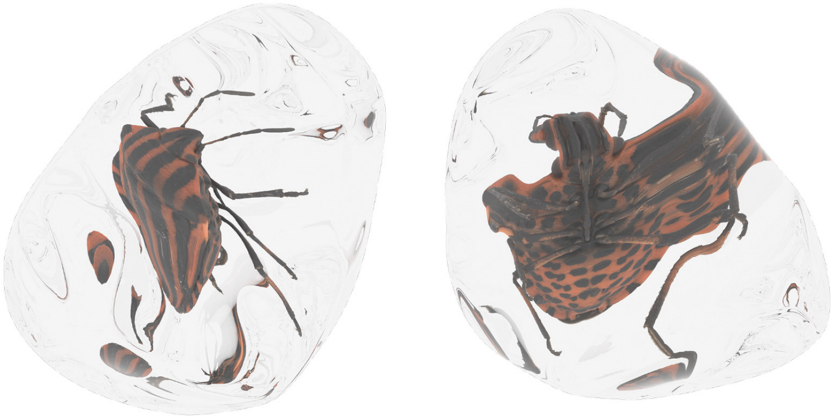


Fig. 17 Two images of the graphosoma immersed in a convex block of epoxy resin.

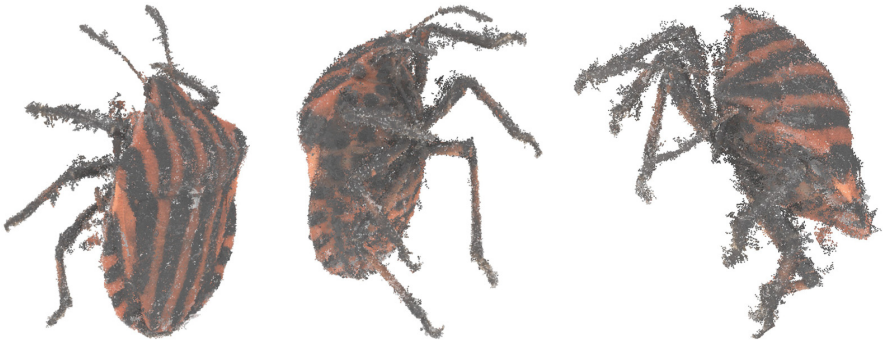


Fig. 18 3D reconstruction of the graphosoma immersed in a convex block of epoxy resin, seen from three angles, using our RMVS solving method from 18 images such as those in Fig. 17.

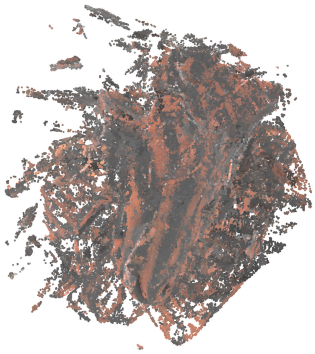


Fig. 19 3D reconstruction by MVS, from 18 images such as those in Fig. 17: the result resembles a random 3D point cloud.

remains true to the original form but is slightly less precise than the reconstruction in Fig. 7. This reduced precision is due to some grazing rays, where the angle i_2 in the Snell–Descartes law (7) approaches $\pi/2$. Consequently, since the derivative of the arcsin function tends towards infinity at 1, this results in calculation inaccuracies for the angle i_1 in Eq. (7). In contrast, Fig. 19 illustrates that neglecting refraction in the reconstruction process yields a result resembling a random 3D point cloud.

In this section, we validated our RMVS solving method only on synthetic images, but purposely. Indeed, to be able to process real images, several additional data are necessary, in addition to the images themselves and the intrinsic parameters of the camera: the shape of the interface (with more or less precision, see Table 3), the poses of the camera and the IoR of the transparent medium.

5 Implementation on Real Images

The primary challenge in applying our 3D reconstruction method to real images lies in estimating camera poses. While the refractive structure-from-motion method suggested in Ref. 53 is an option, it requires prior knowledge of the medium's IoR, which is one of the unknown factors. Additionally, since recovering the 3D shape of the interface is essential, we propose in Sec. 5.1 a simultaneous estimation method for both camera poses and the interface 3D shape. This approach relies on multi-view matching of polyhedron vertices detected in the images and does not rely on the IoR. Consequently, the IoR can be determined a posteriori, as we will discuss in Sec. 5.2.

5.1 Estimating the Camera Poses and the Interface 3D Shape

In this subsection, we detail a method for acquiring the camera poses and the 3D shape of the interface in a shared 3D frame. This involves fixing the camera opposite the object positioned on

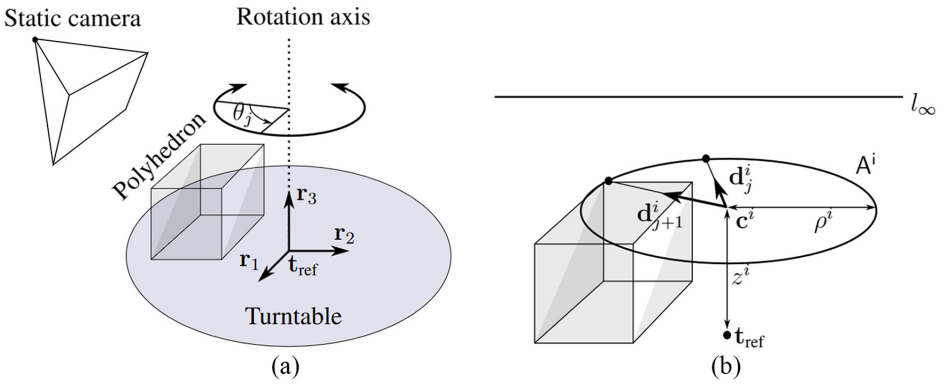


Fig. 20 (a) The acquisition setup involves placing a polyhedron on a turntable and capturing views with a static camera. The origin $\mathbf{t}_{\text{ref}}$ is at the intersection of the table and its rotation axis. The rotation matrix $\mathbf{R}_{\text{ref}}$, having columns $\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$, defines the turntable's pose relative to the camera frame. (b) In consecutive images j and $j+1$, a vertex at a distance ρ^i from the rotation axis, oriented along $\mathbf{d}^i_j$ and $\mathbf{d}^i_{j+1}$, belongs to an ellipse of equation $\mathbf{x}^\top A^i \mathbf{x} = 0$ in the image plane. The imaged center $\mathbf{c}^i$ of this ellipse satisfies the pole-polar relationship $\mathbf{c}^i = [A^i]^{-1}\mathbf{l}_\infty$.

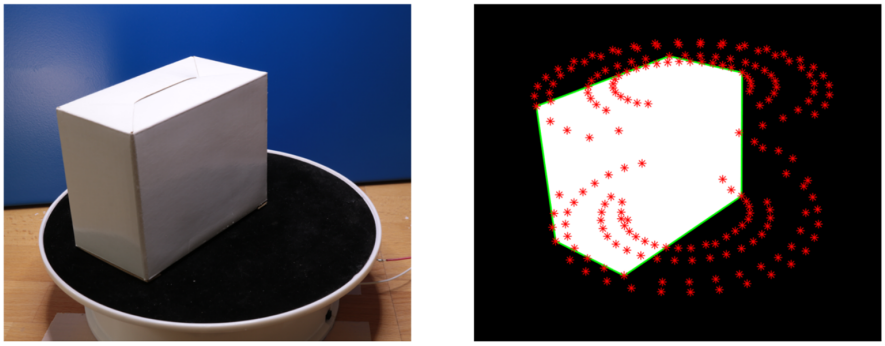


Fig. 21 (a) One of 40 images depicting a parcel on a turntable. In all views, the silhouettes of the parcel, treated as a convex polyhedron, are extracted. (b) The collection V of all silhouette vertices is shown in red.

a rotating table, simulating camera movement around the object. Figures 20(a) and 21(a) illustrate this setup.

We could have concurrently estimated the camera poses and the 3D shape of the interface using shape-from-silhouettes, a technique independent of the IoR of the medium. However, this method struggles with arbitrary shapes, as it computes an enclosing volume by intersecting silhouette back-projections. For satisfactory accuracy, an infinite number of poses are ideal, unless we limit to polyhedral interfaces with a few vertices. For insects in amber, as mentioned in Sec. 1, this is feasible by shaping the amber into a polyhedron with multiple planar faces.

We thus consider a scene with a convex polyhedron of q vertices on a turntable. The vertices' coordinates $\mathbf{X}^i \in \mathbb{R}^3$, $i \in \{1, \ldots, q\}$, are in a 3D frame $\mathcal{R}_{\text{ref}}$ affixed to the turntable. The origin of $\mathcal{R}_{\text{ref}}$ lies at the table's supporting plane and rotation axis intersection, with its first two axes defining the plane and the third as an upward normal vector. A static camera with known intrinsics captures r views of this scene. The homogeneous coordinate vector $\mathbf{x}^i_j \in \mathbb{R}^3$ of the j'th image, $j \in \{1, \ldots, r\}$, of vertex $\mathbf{X}^i$ satisfies the equation

$$\mathbf{x}^i_j \sim \mathrm{P}_j \begin{bmatrix} \mathbf{X}^i \\ 1 \end{bmatrix} \tag{11}$$

with the perspective projection matrix P_j corresponding to the j'th view defined as

$$\mathrm{P}_j = \mathbf{K}[\,\mathbf{R}_{\mathrm{ref}}\mathbf{R}_j|\,\mathbf{t}_{\mathrm{ref}}\,]. \tag{12}$$

where $\mathbf{K}$ represents the calibration matrix. The transformation from $\mathcal{R}_{\mathrm{ref}}$ to the camera frame is specified by $(\mathbf{R}_{\mathrm{ref}}, \mathbf{t}_{\mathrm{ref}})$. The matrix $\mathbf{R}_j$ indicates the table's rotation by an angle θ_j around its axis

$$\mathbf{R}_j = \begin{bmatrix} \cos\theta_j & -\sin\theta_j & 0 \\ \sin\theta_j & \cos\theta_j & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{13}$$

In homogeneous Cartesian coordinates $\mathbf{x} = [x, y, 1]^\top$, an ellipse is represented as $\mathbf{x}^\top A\mathbf{x} = 0$, where A is a symmetric 3×3 matrix under suitable conditions on its coefficients.

The first step is to create a polygonal silhouette for each view and is obtained by simple operations: background subtraction from a reference image, thresholding, morphological processing, extraction, and simplification of the convex hull to get the silhouette vertices. We then assemble the collection V of all these vertices. An example of extracted silhouette vertices V, highlighted in red, is shown in Fig. 21(b).

The second step requires robustly partitioning V into subsets on common ellipses, representing parallel circular trajectories in 3D space. The partition size, corresponding to the polyhedron's vertices, is unknown. A partitioning solution is detailed in Ref. 2, utilizing the parallelism of vertices' trajectories. The images of circular points (ICP)[58] of the turntable, two complex conjugate vectors in $\mathbb{C}^3$ denoted as $\mathbf{h}_1 \pm \mathrm{i}\mathbf{h}_2$, are estimated along with correspondences. Details can be found in Ref. 2.

With the ICP $\mathbf{h}_1 \pm \mathrm{i}\mathbf{h}_2$, correspondences $\{\mathbf{x}_j^i\}$ and calibration matrix K known, the problem is to determine the vertices' positions and the polyhedron's poses in the camera frame. Specifically, this includes calculating the rotation matrix $\mathbf{R}_{\mathrm{ref}}$, the translation vector $\mathbf{t}_{\mathrm{ref}}$, the 3D coordinates of the vertices $\{\mathbf{X}^i\}_{i\in\{1,\ldots,q\}}$, and the angles $\{\theta_j\}_{j\in\{1,\ldots,r\}}$. Matrix $\mathbf{R}_{\mathrm{ref}}$ and vector $\mathbf{t}_{\mathrm{ref}}$ are computed using the method from Ref. 59.

The rotation angle θ_j of the turntable in view number j is measured from a reference position θ_1 as follows:

$$\theta_j = \sum_{k=1}^{j-1} \theta_{k,k+1}, \tag{14}$$

where $\theta_{k,k+1}$ represents the rotation angle between two consecutive acquisitions k and $k+1$. Its value is determined as the median cosine of the estimated angles from the visible vertices

$$\theta_{k,k+1} = \mathrm{acos}(\underset{i\in\mathcal{D}_k}{\mathrm{median}}\{\mathbf{d}_k^{i\top}\mathbf{d}_{k+1}^i\}), \tag{15}$$

where $\mathcal{D}_k \subset \{1, \ldots, q\}$ represents the set of vertex indices detected in both the k^{th} and $(k+1)^{\mathrm{th}}$ images. The unit vectors $\mathbf{d}_k^i$ and $\mathbf{d}_{k+1}^i$ point towards the images by H^{-1} of the corresponding points $\mathbf{x}_k^i$ and $\mathbf{x}_{k+1}^i$, where $H = [\mathbf{h}_1\,\mathbf{h}_2\,*]$, in the sequential views k and $k+1$, specifically

$$\mathbf{d}_k^i = \frac{f(H^{-1}\mathbf{x}_k^i) - f(H^{-1}\mathbf{c}^i)}{\|f(H^{-1}\mathbf{x}_k^i) - f(H^{-1}\mathbf{c}^i)\|}, \tag{16}$$

and likewise for $\mathbf{d}_{k+1}^i$. In Eq. (16), $f([u, v, w]^\top) = [u/w, v/w]^\top$, and $\mathbf{c}^i$ is the homogeneous coordinate vector of the image of the trajectory's center, assumed circular, of the vertex number i, and derived from the pole-polar relation $\mathbf{c}^i = [\mathrm{A}^i]^{-1}\mathbf{l}_\infty$. Here $\mathbf{l}_\infty$ is the vanishing line vector of the table plane, and is the cross-product $\mathbf{l}_\infty = \mathbf{h}_1 \times \mathbf{h}_2$ and A^i the matrix of the ellipse image of vertex number i's trajectory [see Fig. 20(b)]. The table rotation during acquisition is assumed to be counterclockwise. The $\theta_{k,k+1}$ values are supposed to be between 0 and 180 deg.

At this point, all camera poses are known, and the 3D coordinates of the vertices $\{\mathbf{X}^i\}$ are obtained by triangulating the correspondences $\{\mathbf{x}_j^i\}$. Both are further refined through a bundle adjustment minimizing the Euclidean distances between the correspondences and their reprojections.

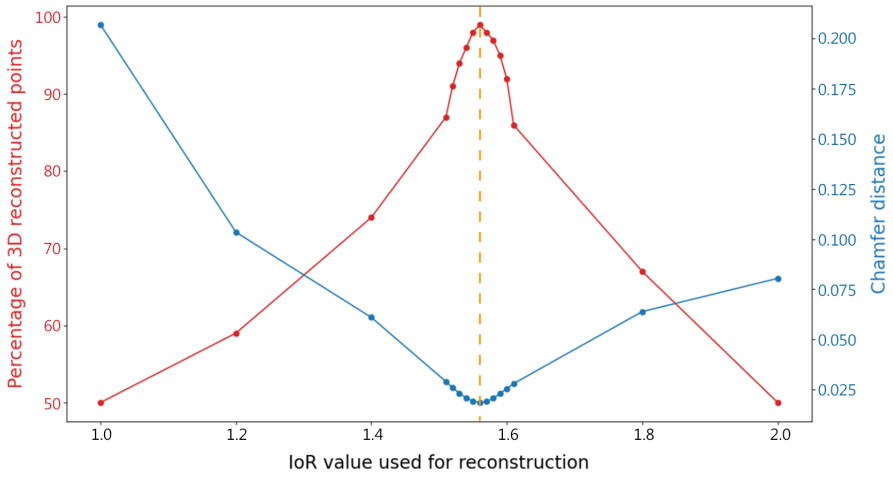


Fig. 22 Percentage of reconstructed 3D points (red) and Chamfer distance (blue) variation with the index of refraction (IoR) of the refractive medium surrounding the graphosoma. When simulating the images, the IoR used ($n_2 = 1.56$) aligns exactly with the peak of the red curve and the lowest CD, validating our proposed criterion.

5.2 Validation on Real Images

Estimating the index of refraction can typically be done using a dedicated instrument known as a refractometer. However, we suggest an alternative estimation method in this subsection, leveraging the joint estimation of camera poses and the 3D shape of the interface, as outlined in the previous subsection. Specifically, our RMVS solving method, detailed in Sec. 3 and tested on synthetic images in Sec. 4, can be applied with varying IoR values, ideally within a range close to its "plausible" value. The challenge lies in identifying a sufficiently discriminating criterion for this estimation. As illustrated in Figs. 22 and 23, the number of effectively reconstructed 3D points serves as such a criterion, since it shows the maximum number of points for the exact IoR and the lowest Chamfer distance score for the associated reconstruction.

We can now apply the complete RMVS solving pipeline to real data, provided the interface is polyhedral. Figures 24 and 25 display tests conducted on two epoxy-resin parallelepipeds, containing a beetle and a grasshopper, respectively.

The 3D reconstructions of these two insects, shown in Figs. 26 and 27, reveal a noticeably better reconstruction of the beetle compared to the grasshopper. This disparity primarily stems from the resin block containing the grasshopper, which fails to fully meet the assumptions underlying our RMVS solving method. First, one of the block's faces is not as planar as required.

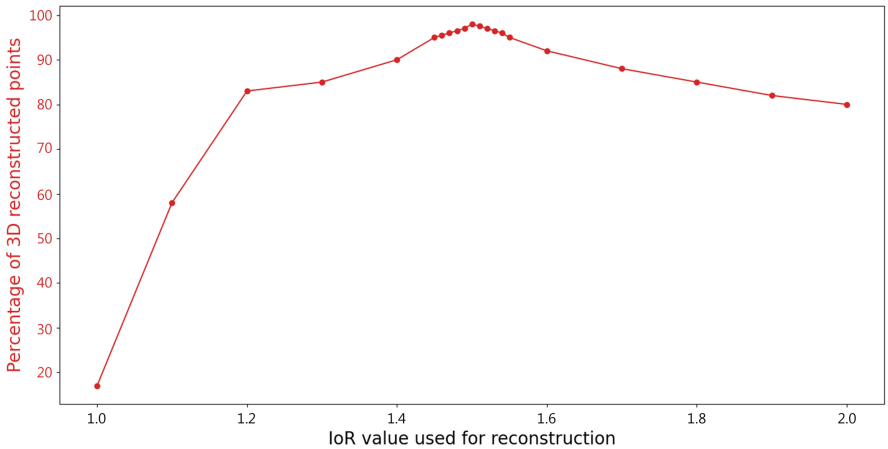


Fig. 23 Evolution of the percentage of reconstructed 3D points, in function on the IoR of the refractive medium in which the real beetle from Fig. 25 is immersed. The maximum of this curve gives us an estimate of the IoR equal to $n_2 = 1.50$.

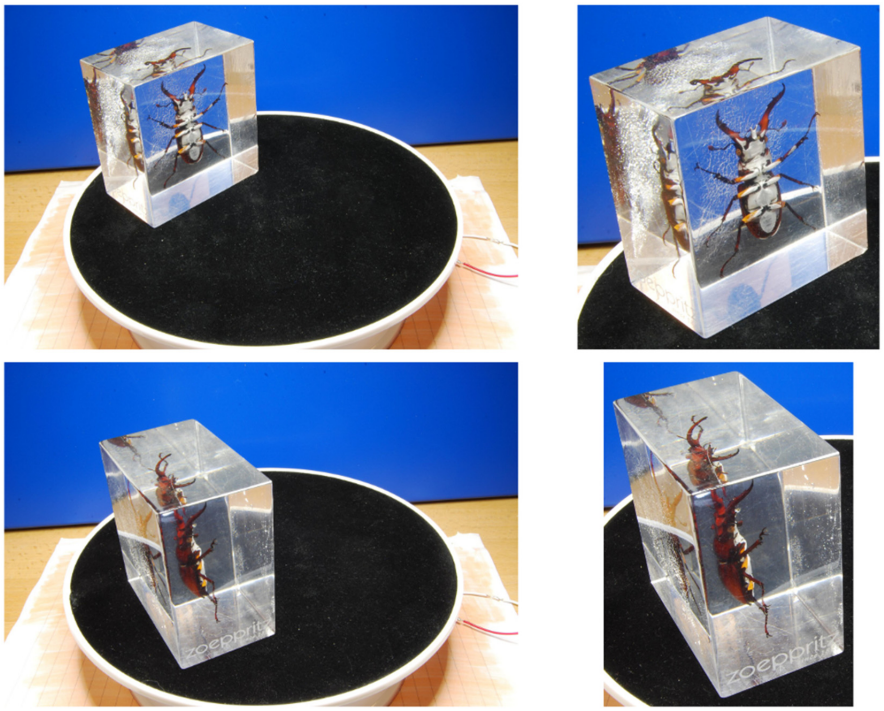


Fig. 24 Two real images of a beetle immersed in a parallelepipedic block of resin, placed on a turntable, and zooms on the block.

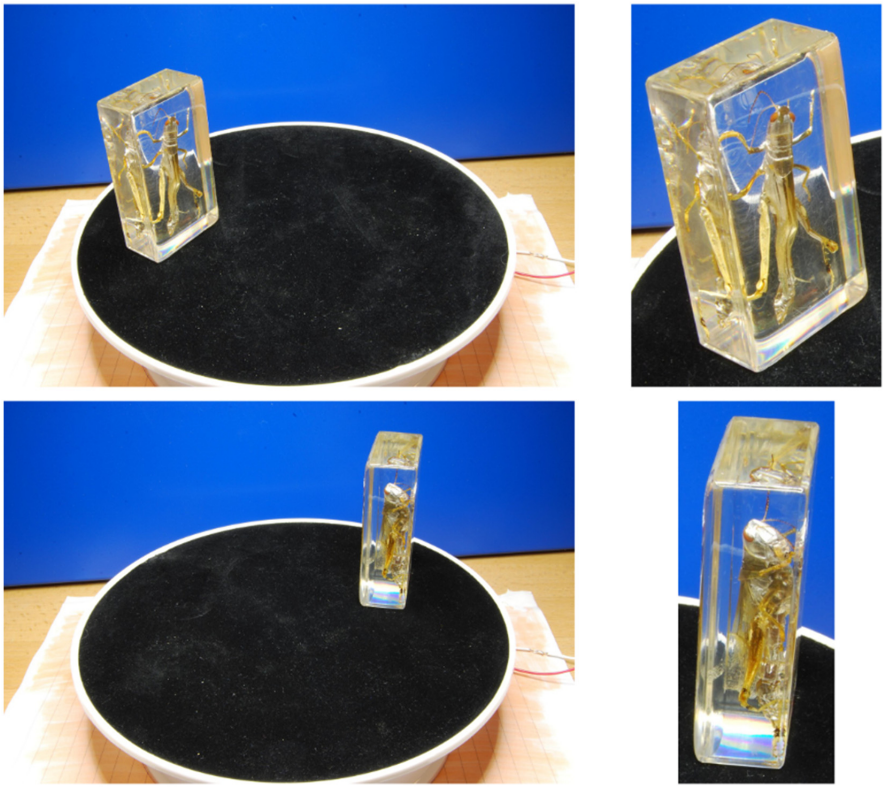


Fig. 25 Two real images of a grasshopper immersed in a parallelepipedic block of resin, placed on a turntable, and zooms on the block.

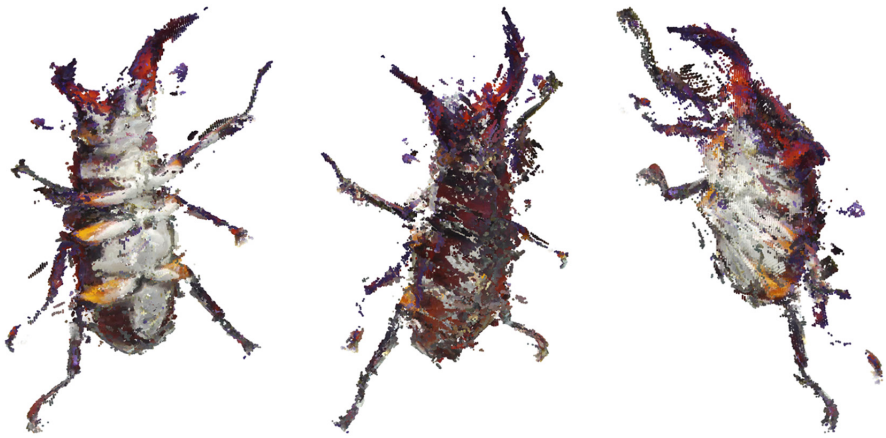


Fig. 26 3D reconstruction of the beetle from 24 images, such as those in Fig. 24, using our RMVS method.

Fig. 27 3D reconstruction of the grasshopper from 24 images, such as those in Fig. 25, using our RMVS method.

Fig. 28 Comparison of our RMVS method, tested on the same 24 real images of the beetle (see Fig. 24). We used either the IoR value estimated by the method illustrated in Fig. 23 ($n_2 = 1.50$), or a slightly overvalued IoR ($n_2' = 1.56$). The first result is obviously more accurate.

Second, the resin exhibits layered structure visible to the naked eye, indicating that light rays within the refractive medium may not travel in perfectly straight lines. Additionally, a significant distinction between the results in Figs. 26 and 27 lies in the insects themselves. Certain parts of the grasshopper's body appear somewhat translucent, challenging a fundamental premise of the MVS technique and its variants, which is the assumption that the surface being reconstructed should be opaque and Lambertian.

A final experiment with the beetle images aimed to qualitatively assess how using an incorrect IoR value affects the reconstruction outcome. Figure 28 displays two 3D reconstructions of the beetle, each derived using different IoR values: the right image, produced with a slightly overvalued IoR ($n_2' = 1.56$), is noticeably less accurate than the left image, where the IoR ($n_2 = 1.50$) was determined using the previously described method (refer to Fig. 23).

6 Conclusion and Perspectives

In this paper, we adapted the MVS technique for objects immersed in a refractive medium. Given that refraction distorts images, it is crucial to model light ray paths accordingly. We introduced a fully discrete RMVS solving method, with promising initial results on real data, despite several challenges before it becomes a practical tool for entomologists.

A future direction involves assessing the RMVS method's robustness against imperfect knowledge of interface geometry, such as non-planar polyhedron faces. The use of UV, IR, and polarized lights could also help us to constrain the interface geometry and to reduce some refraction/reflection effects. Another area for development is automating the detection of silhouettes within the refractive medium. Neural methods, as suggested by Ref. 60, could be a solution.

Furthermore, methods using differentiable rendering, like ReNeuS, are increasingly important. We were unfortunately unable to test ReNeuS as its code is not publicly available (and it does consider only boxed-shaped media). However, such approaches remain a short-term goal, whether to solve the RMVS problem addressed in this paper or to solve photometric stereo under refraction.[36]

A longer-term goal is to develop a pipeline for acquiring and processing data, particularly prehistoric insects trapped in amber. Overcoming numerous challenges is necessary, as the poor result in Fig. 27 is due to both the resin block and the contained object not fully meeting our RMVS method's assumptions. The pipeline needs to be robust against predictable flaws, particularly when the index of refraction is not uniform. Additionally, even under the Lambertian assumption, coloration in the refractive medium can alter a 3D point's appearance across images due to varying light travel distances. Focus blur, a small-scale challenge we have overlooked also needs consideration. Addressing these factors should enhance the quality of our results.

Code and Data Availability

Our real/synthetic data and code will be available on demand.

Acknowledgments

Robin Bruneau's doctoral student fellowship is funded by the Danish project UCPH Data+ PHYLORAMA. Baptiste Brument's doctoral student fellowship is funded by the French Ministry of Higher Education and Research. This work was partly funded by the French National Research Agency through the LabCom project ALICIA-Vision and the Inclusive Museum Guide project (Grant No. ANR-20-CE38-0007).

References

1. M. Cassidy et al., "Refractive multi-view stereo," in *Proc. Int. Conf. 3D Vis*. (2020).
2. B. Brument et al., "A shape-from-silhouette method for 3D-reconstruction of a convex polyhedron," in *Proc. Quality Control by Artif. Vis. Conf*. (2023).
3. H. G. Maas, "New developments in multimedia photogrammetry," in Optical 3-D Measurement Techniques III (1995).
4. T. Łuczyński, M. Pfingsthorn, and A. Birk, "Image rectification with the pinax camera model in underwater stereo systems with verged cameras," in *OCEANS 2017*, pp. 1–7 (2017).

5. P. Agrafiotis et al., "Correcting image refraction: towards accurate aerial image-based bathymetry mapping in shallow waters," *Remote Sens.* **12**(2), 322 (2020).
6. X. Wu and X. Tang, "Accurate binocular stereo underwater measurement method," *Int. J. Adv. Rob. Syst.* **16**(5), 1729881419864468 (2019).
7. P. Agrafiotis et al., "DepthLearn: learning to correct the refraction on point clouds derived from aerial imagery for accurate dense shallow water bathymetry based on SVMs-fusion with LiDAR point clouds," *Remote Sens.* **11**(19), 2225 (2019).
8. K. Ichimaru and H. Kawasaki, "Underwater stereo using refraction-free image synthesized from light field camera," in *Proc. IEEE Int. Conf. Image Process.*, pp. 1039–1043 (2019).
9. C. Zhang et al., "On-site calibration of underwater stereo vision based on light field," *Opt. Lasers Eng.* **121**, 252–260 (2019).
10. D. H. Lee, I.-S. Kweon, and R. Cipolla, "A biprism-stereo camera system," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, Vol. **1** (1999).
11. A. Yamashita, Y. Shirane, and T. Kaneko, "Monocular underwater stereo – 3D measurement using difference of appearance depending on optical paths," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. and Syst.*, pp. 3652–3657 (2010).
12. Z. Chen et al., "Depth from refraction using a transparent medium with unknown pose and refractive index," *Int. J. Comput. Vis.* **102**(1–3), 3–17 (2013).
13. C. Gao and N. Ahuja, "A refractive camera for acquiring stereo and super-resolution images," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 2316–2323 (2006).
14. N. J. W. Morris, "Image-based water surface reconstruction with refractive stereo," Master's thesis, Department of Computer Science, University of Toronto (2004).
15. N. J. W. Morris and K. N. Kutulakos, "Dynamic refraction stereo," *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(8), 1518–1531 (2011).
16. M. Ben-Ezra and S. K. Nayar, "What does motion reveal about transparency?" in *Proc. IEEE Int. Conf. Comput. Vis.*, Vol. **2**, pp. 1025–1032 (2003).
17. Z. Li, Y.-Y. Yeh, and M. Chandraker, "Through the looking glass: neural 3D reconstruction of transparent shapes," in *Proc. IEEE/CVF Conf. Comput. Vis. and Pattern Recognit.*, pp. 1262–1271 (2020).
18. M. Shao et al., "Polarimetric inverse rendering for transparent shapes reconstruction," arXiv:2208.11836 (2022).
19. T. Murase et al., "A photogrammetric correction procedure for light refraction effects at a two-medium boundary," *Photogramm. Eng. Remote Sens.* **74**(9), 1129–1136 (2008).
20. A. S. Woodget, J. T. Dietrich, and R. T. Wilson, "Quantifying below-water fluvial geomorphic change: the implications of refraction correction, water surface elevations, and spatially variable error," *Remote Sens.* **11**(20), 2415 (2019).
21. B. Cao, R. Deng, and S. Zhu, "Universal algorithm for water depth refraction correction in through-water stereo remote sensing," *Int. J. Appl. Earth Observ. Geoinf.* **91**, 102108 (2020).
22. P. Sturm, "Multi-view geometry for general camera models," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, Vol. **1**, pp. 206–212 (2005).
23. V. Chari and P. Sturm, "Multi-view geometry of the refractive plane," in *Proc. Br. Mach. Vis. Conf.*, pp. 1–11 (2009).
24. C. Chen et al., "Three-dimensional reconstruction from a fringe projection system through a planar transparent medium," *Opt. Express* **30**(19), 34824–34834 (2022).
25. A. Jordt-Sedlazeck and R. Koch, "Refractive calibration of underwater cameras," *Lect. Notes Comput. Sci.* **7576**, 846–859 (2012).
26. A. Jordt-Sedlazeck, D. Jung, and R. Koch, "Refractive plane sweep for underwater images," in *Proc. German Conf. Pattern Recognit.*, pp. 333–342 (2013).
27. A. Jordt, K. Köser, and R. Koch, "Refractive 3D reconstruction on underwater images," *Methods Oceanogr.* **15**–**16**, 90–113 (2016).
28. S. Haner and K. Åström, "Absolute pose for cameras under flat refractive interfaces," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 1428–1436 (2015).
29. M. Castillón et al., "State of the art of underwater active optical 3D scanners," *Sensors* **19**(23), 5161 (2019).
30. M. Alterman and Y. Y. Schechner, "3D in natural random refractive distortions," *Proc. SPIE* **9867**, 98670B (2016).
31. M. Alterman, Y. Y. Schechner, and Y. Swirski, "Triangulation in random refractive distortions," *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(3), 603–616 (2017).
32. J. Xiong and W. Heidrich, "In-the-wild single camera 3D reconstruction through moving water surfaces," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pp. 12558–12567 (2021).
33. C. Tsiotsios et al., "Backscatter compensated photometric stereo with 3 sources," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 2251–2258 (2014).

34. S. G. Narasimhan et al., "Structured light in scattering media," in *Proc. IEEE Int. Conf. Comput. Vis.*, Vol. **1**, pp. 420–427 (2005).
35. H. Fan et al., "Refractive laser triangulation and photometric stereo in underwater environment," *Opt. Eng.* **56**(11), 113101 (2017).
36. Y. Quéau et al., "On photometric stereo in the presence of a refractive interface," in *Proc. Int. Conf. Scale Space and Variational Methods in Comput. Vis.*, pp. 691–703 (2023).
37. S. Liu et al., "Soft rasterizer: a differentiable renderer for image-based 3D reasoning," in *Proc. IEEE/CVF Int. Conf. on Comput. Vis.*, pp. 7708–7717 (2019).
38. J. Lyu et al., "Differentiable refraction-tracing for mesh reconstruction of transparent objects," *ACM Trans. Graph.* **39**(6), 1–13 (2020).
39. J. Munkberg et al., "Extracting triangular 3D models, materials, and lighting from images," in *Proc. IEEE/CVF Conf. Comput. Vis. and Pattern Recognit.*, pp. 8280–8290 (2022).
40. M. Nimier-David et al., "Mitsuba 2: a retargetable forward and inverse renderer," *ACM Trans. Graph.* **38**(6), 1–17 (2019).
41. W. Jakob et al., "Dr. Jit: a just-in-time compiler for differentiable rendering," *ACM Trans. Graph.* **41**(4), 1–19 (2022).
42. K. Yan et al., "Efficient estimation of boundary integrals for path-space differentiable rendering," *ACM Trans. Graph.* **41**(4), 1–13 (2022).
43. M. Bemana et al., "Eikonal fields for refractive novel-view synthesis," in *Proc. ACM SIGGRAPH Conf.*, pp. 1–9 (2022).
44. T. Fujitomi et al., "LB-NERF: light bending neural radiance fields for transparent medium," in *Proc. IEEE Int. Conf. Image Process.*, pp. 2142–2146 (2022).
45. P. Wang et al., "Neus: learning neural implicit surfaces by volume rendering for multi-view reconstruction," arXiv:2106.10689 (2021).
46. Y. Wang et al., "NeuS2: fast learning of neural implicit surfaces for multi-view reconstruction," in *Proc. IEEE/CVF Int. Conf. on Comput. Vis.* (2023).
47. J. Tong et al., "Seeing through the glass: neural 3D reconstruction of object inside a transparent container," in *Proc. IEEE/CVF Conf. Comput. Vis. and Pattern Recognit.*, pp. 12555–12564 (2023).
48. Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(8), 1362–1376 (2010).
49. L. Kang, L. Wu, and Y.-H. Yang, "Two-view underwater structure and motion for cameras under flat refractive interfaces," *Lect. Notes Comput. Sci.* **7575**, 303–316 (2012).
50. A. Agrawal et al., "A theory of multi-layer flat refractive geometry," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 3346–3353 (2012).
51. Y. Furukawa and C. Hernández, "Multi-view stereo: a tutorial," *Found. Trends Comput. Graph. Vis.* **9**(1-2), 1–148 (2015).
52. M. Goesele, B. Curless, and S. M. Seitz, "Multi-view stereo revisited," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.*, pp. 2402–2409 (2006).
53. F. Chadebecq et al., "Refractive two-view reconstruction for underwater 3D vision," *Int. J. Comput. Vis.* **128**(5), 1101–1117 (2020).
54. Y. Nehmé et al., "Textured mesh quality assessment: large-scale dataset and deep learning-based quality metric," arXiv:2202.02397 (2023).
55. E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik* **1**, 269–271 (1959).
56. C. Griwodz et al., "AliceVision meshroom: an open-source 3D reconstruction pipeline," in *Proc. 12th ACM Multimedia Syst. Conf.*, pp. 241–247 (2021).
57. J.-D. Durou, M. Falcone, and M. Sagona, "Numerical methods for shape-from-shading: a new survey with benchmarks," *Comput. Vis. Image Underst.* **109**(1), 22–43 (2008).
58. R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press (2004).
59. P. Sturm, "Algorithms for plane-based pose estimation," in *Proc. IEEE Conf. Comput. Vis. and Pattern Recognit.* (2000).
60. A. Kirillov et al., "Segment anything," arXiv:2304.02643 (2023).

Robin Bruneau is a PhD student in computer science at DIKU and IRIT. His research focuses on 3D reconstruction under refraction. This has led him to confront the classical methods of 3D reconstruction, as well as neural methods, while bringing an understanding of refraction to solve its own project.

Baptiste Brument is a PhD student in computer vision. He is working on 3D reconstruction from optical imagery. His recent research has centered on addressing the multi-view photometric stereo problem, employing neural rendering techniques.

Lilian Calvet is a senior researcher in computer vision with the ROCS team at the University of Zurich and Balgrist University Hospital. He holds a PhD in computer vision from the University of Toulouse, France. His primary research interests include inverse problems and geometric methods in imaging. Currently, his work focuses on 3D reconstruction, pose estimation and image-based registration, and their application to computer-assisted surgery.

Matthew Cassidy is a computer vision engineer in Toulouse who worked at IRIT for an internship about 3D reconstruction under refraction.

Jean Mélou is a postdoctoral researcher at IRIT laboratory, in Toulouse, where he obtained his PhD in 2020. His research focuses on 3D reconstruction, in particular the fusion of different types of approaches, such as multi-view, photometric, or neural network methods. His research finds applications in different fields, such as entomology, post-production, or archaeology.

Yvain Quéau is a CNRS research fellow in the Image team of GREYC Laboratory in Caen (France). He obtained his PhD in computer science from the Institut National Polytechnique de Toulouse in 2015 and was then a post-doc in Daniel Cremer's group in TU Munich (Germany). His primary research interest is inverse problems in computer vision and their solving using variational methods and deep learning.

Jean-Denis Durou received his PhD in computer science from the Université Paris 11 in 1993, and the HDR ("habilitation à diriger les recherches") from the Université Toulouse 3 in 2007. He is an assistant professor at the Université de Toulouse and a member of the REVA team at the IRIT. His main research interest is 3D vision. He is more specifically interested in photometric 3D reconstruction, i.e., shape-from-shading and photometric stereo.

François Lauze is an associate professor at the Department of Computer Science, University of Copenhagen. He holds a PhD in mathematics, the University of Nice-Sophia-Antipolis and a PhD in Image Analysis, IT University of Copenhagen. His main research interests are inverse problems and geometric methods in imaging. His current work principally focuses on 3D reconstruction on one side and geometry-flavored deep learning on the other side.

Chapter C

Third paper

On Photometric Stereo in the Presence of a Refractive Interface

Yvain Quéau[1(✉)], Robin Bruneau[2,3], Jean Mélou[2], Jean-Denis Durou[2], and François Lauze[3]

[1] Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France
yvain.queau@ensicaen.fr
[2] IRIT, UMR CNRS 5505, Toulouse, France
[3] DIKU, Copenhagen University, Copenhagen, Denmark

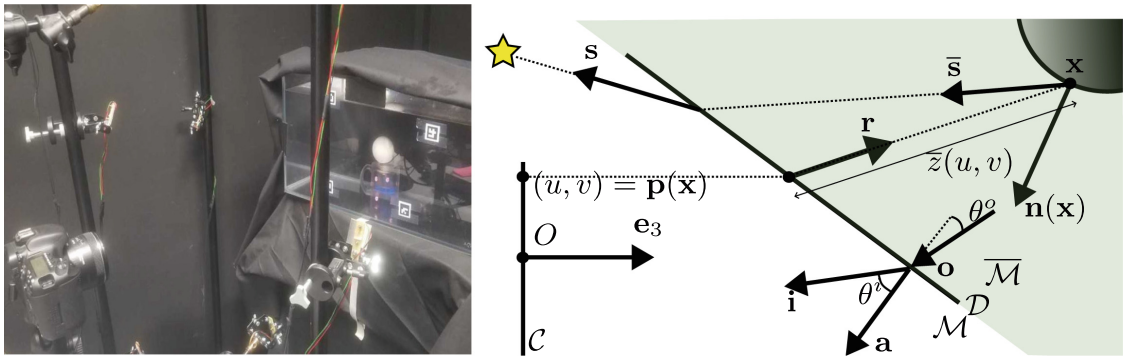
Abstract. We conduct a discussion on the problem of 3D-reconstruction by calibrated photometric stereo, when the surface of interest is embedded in a refractive medium. We explore the changes refraction induces on the problem geometry (surface and normal parameterization), and we put forward a complete image formation model accounting for refracted lighting directions, change of light density and Fresnel coefficients. We further show that as long as the camera is orthographic, lighting is directional and the interface is planar, it is easy to adapt classic methods to take into account the geometric and photometric changes induced by refraction. Moreover, we show on both simulated and real-world experiments that incorporating these modifications of PS methods drastically improves the accuracy of the 3D-reconstruction.

1 Introduction

Photometric stereo (PS) is a 3D computer vision technique which was pioneered by Woodham in the late 70s [27]. It aims at inferring the shape of an opaque surface from a series of images captured under the same viewing angle, but varying illumination. Compared to other 3D-reconstruction techniques, PS excels at recovering the thinnest geometric variations (high-frequency information given by surface normals), and it is the only photographic 3D-reconstruction method which is also able to infer the reflectance of the surface. Such properties are essential in applications such as relighting or cultural heritage artifacts digitization.

However, a fundamental assumption in PS is that the light sources, the camera and the pictured surface all lie in the same homogeneous medium – usually the air. In the present paper, we revisit PS in the presence of a refractive interface i.e., when the camera and the light sources both lie in one homogeneous medium, while the surface is immersed in another homogoneous medium with a different index of refraction (pure water, glass, alcohol, etc.). This particular setting finds applications, for instance, in underwater imaging (Fig. 1a) or in the digitization of natural historic museal objects preserved in amber or alcohol.

L. Calatroni et al. (Eds.): SSVM 2023, LNCS 14009, pp. 691–703, 2023.
https://doi.org/10.1007/978-3-031-31975-4_53



(a) (b)

Fig. 1. We discuss the problem of recovering through photometric stereo the 3D-shape and the albedo of a surface immersed in a refractive medium, as in (a) where a white sphere is immersed in an aquarium filled with pure water. In particular, we show how to adapt classic PS methods when the lighting is directional, the camera is orthographic and the interface is planar, as illustrated in the sketch (b) which summarizes our notations. Therein, given a plane $\mathcal{D}$ with normal $\mathbf{a}$, Snell's law (2.3) gives the relation between an incident ray $\mathbf{i}$ in medium $\mathcal{M}$, and the refracted one $\mathbf{o}$ in medium $\overline{\mathcal{M}}$. Even if the camera is orthographic, a point $\mathbf{x}$ on the surface projects non-orthogonally onto the image plane $\mathcal{C}$: a pixel (u, v) first deprojects onto $\mathcal{D}$ along the viewing direction $\mathbf{e}_3$, and then travels the distance $\bar{z}(u, v)$ along the refracted viewing direction $\mathbf{r}$. Besides, the effective lighting direction $\bar{\mathbf{s}}$ differs from the direction $\mathbf{s}$ which is calibrated outside the refractive medium.

The difference with classic PS lies in the presence of an interface between the two media, which have different indices of refraction. Refraction will have profound consequences in 3D shape recovery techniques, as it modifies the geometry of image acquisition, and light direction and density will be changed as well (this is, after all, the principle behind a lot of lensing effects). While these points are well-understood by designers of optical systems, either to use them or for limiting some of their undesirable consequences, they have seldom been investigated from the photometric shape recovery side.

Assumptions and Contributions. We address the PS problem, in the presence of a *Lambertian* surface (specularities are viewed as outliers) embedded in a *homogeneous* refractive medium with *known geometry*, imaged in the *visible spectrum*. After reviewing related works in Sect. 2, we explore the impact of a *planar* (but not necessarily fronto-parallel) refractive interface on the geometry of PS under *orthographic projection* in Sect. 3. In Sect. 4, we derive a complete image formation model for this case, under *directional lighting calibrated outside the refractive medium*. This model accounts for refraction of lighting directions, attenuation of lighting densities, and Fresnel coefficients. Then, we discuss in Sect. 5 the inversion of this model by adapting classic PS algorithms. Eventually, in Sect. 6 we draw our conclusions, and mention possible extensions of our work to more complicated setups (pinhole camera, non-directional lighting, non-planar interface, and light absorption).

2 Background

Photometric Stereo. In the traditional PS setup, the pictured surface $\mathcal{S}$ is assumed Lambertian i.e., it reflects light diffusively, as the reflectance at $\mathbf{x} \in \mathcal{S}$ is characterized by the albedo $\rho(\mathbf{x}) \in [0, 1]$. Let us consider a surface lit by a single, known point light source at infinity (calibrated directional lighting) represented by unit direction $\mathbf{s} \in \mathbb{R}^3$ and density $\varphi > 0$, and denote $\mathbf{n}(\mathbf{x}) \in \mathbb{R}^3$ the unit outward normal to the surface at $\mathbf{x}$. Then, the measured brightness at pixel $(u, v) = \mathbf{p}(\mathbf{x})$, which is the projection of the surface point $\mathbf{x}$ onto the camera image plane, is $I(\mathbf{p}) \propto \varphi \max\{0, \rho(\mathbf{x})\mathbf{n}(\mathbf{x})^\top \mathbf{s}\}$, with the proportionality constant independent of $\mathbf{x}$. Omitting the $\max\{\}$ operator, which models self-shadows (they are usually dealt with robust estimators), integrating the proportionality coefficient into the albedo (which can be normalized a posteriori), and considering $k \geq 3$ light sources yields the following image formation model:

$$I_i(\mathbf{p}) = \varphi_i \, \rho(\mathbf{x}) \, \mathbf{n}(\mathbf{x})^\top \mathbf{s}_i, \quad i \in \{1, \ldots, k\}. \tag{2.1}$$

This model can be inverted as long as the light directions $\mathbf{s}_i$ are non-coplanar, so as to compute the Lambertian reflectance $\rho(\mathbf{x})$ and the surface normal $\mathbf{n}(\mathbf{x})$ for each $\mathbf{p}$. This approach can also be extended to non-Lambertian reflectance and uncalibrated lighting, for instance by resorting to deep neural networks [7].

Surface Parameterization. The surface $\mathcal{S}$ is parameterized as $(u, v) \mapsto \mathbf{x}(u, v) = \mathcal{S}(u, v)$ and its normal at $\mathbf{x}$ is written as

$$\mathbf{n}(\mathbf{x}) = \pm \frac{\mathcal{S}_u \times \mathcal{S}_v}{|\mathcal{S}_u \times \mathcal{S}_v|}, \tag{2.2}$$

where $\mathcal{S}_u$ and $\mathcal{S}_v$ are the partial derivatives of $\mathcal{S}$, and where the $\pm$ ambiguity is resolved by taking arbitrarily the normal oriented towards the camera. Once the normal field $\mathbf{n}(\mathbf{x})$ is estimated, retrieving $\mathcal{S}$ then comes down to a 2D integration problem, for which various solutions exist [20]. The parameterization $\mathcal{S}$ is a right-inverse to the projection: $\mathbf{p}(\mathcal{S}(u, v)) = (u, v)$. It is constrained by the form that $\mathbf{p}$ takes (orthographic projection, perspective projection, etc.), and this has of course important consequences on the integration process.

Refraction. The index of refraction (IoR) n of a material is the ratio c/v of the speed of light in vacuum and the velocity in the medium. Snell's laws assert that 1) the normal $\mathbf{a}$ to the interface, the incident light direction $\mathbf{i}$ and the refracted light direction $\mathbf{o}$ are coplanar; and 2) the refracted and incident angles satisfy the relation $n \sin\theta^i = \overline{n} \sin\theta^o$, with n the IoR of the first medium, $\overline{n}$ the IoR of the second one, θ^i the angle between $\mathbf{i}$ and $\mathbf{a}$, and θ^o the angle between $\mathbf{a}$ and $\mathbf{o}$ (see Fig. 1b). In vectorial form [14], defining $\mu = n/\overline{n}$:

$$\mathbf{o} = \mathrm{Snell}_\mu^{\mathbf{a}}(\mathbf{i}) = \mu \, \mathbf{i} + \left(\sqrt{1 - \mu^2 \, (1 - (\mathbf{i}^\top \mathbf{a})^2)} - \mu \, (\mathbf{i}^\top \mathbf{a}) \right) \mathbf{a}. \tag{2.3}$$

Refractive 3D-Vision. Snell's law (2.3) of refraction has been considered in few 3D-vision contexts. For instance, the epipolar geometry theory has been extended to the case where the camera and the surface are separated by a refractive plane [5]. This constitutes the basis for the development of refractive structure-from-motion algorithms [4,13]. Multi-view stereo in the presence of a refractive interface has also been recently explored [3,6,11]. In the photometric stereo context, underwater imaging has attracted some attention [10,16,17,25,26]. These works focus mostly on light absorption, which occurs when scattering is involved (inhomogeneous medium such as murky water) or in near-infrared imaging. Yet, other refraction effects (e.g., change of incident light direction and density, and of surface parameterization) are neglected. For instance, it is usually assumed that all the sources have the same relative intensity, and that their directions can be obtained using a calibration target immersed in the medium. Yet, even if all the sources outside the refractive medium have exactly the same intensity, the refractive interface will induce luminous fluxes with different densities (see Sect. 4). Therefore, it would be more convenient to calibrate light directions and densities outside the refractive medium, and account for refraction within the image formation model. This has been achieved in [18] but only for a fronto-parallel interface and with a somehow naive numerical solution, and in [9] but by relying on laser triangulation. Instead, the present paper aims at modeling and evaluating the effects of a refractive interface with arbitrary orientation on shape recovery by pure PS, and at providing an efficient numerical solution by adapting state-of-the-art algorithms.

3 Geometry of Refractive PS

Notations. As illustrated in Fig. 1b, we work in $\mathbb{R}^3$ with its canonical frame $(O, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$, where O is the camera's principal point, $\mathbf{e}_3$ is the optical axis direction and $\mathcal{C} := \mathbf{e}_3^\perp$ is the image plane. A generic point in $\mathbb{R}^3$ is denoted by $\mathbf{x}$, while $\mathbf{p}(\mathbf{x}) = (u, v)^\top$ denote the 2D coordinates of its conjugate pixel in the frame $(O, \mathbf{e}_1, \mathbf{e}_2)$. The projection from $\mathbb{R}^3 \to \mathbb{R}^2$ keeping the first two coordinates is represented by the matrix $\mathbf{\Pi} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$, whose transpose is the canonical injection $\mathbb{R}^2 \to \mathbb{R}^3$. The interface plane $\mathcal{D}$ is given by the equation $\mathbf{a}^\top \mathbf{x} + \alpha = 0$, $\alpha \in \mathbb{R}$, where $\mathbf{a} = (a_1, a_2, a_3)^\top \in \mathbb{S}^2$ is a known unit normal vector to $\mathcal{D}$ ($\mathbb{S}^2$ being the unit sphere of $\mathbb{R}^3$), oriented towards the camera ($\mathbf{a}^\top \mathbf{e}_3 \leq 0$). We assume that $\mathbf{a}^\top \mathbf{e}_3 \neq 0$. The medium containing the camera is located in $\mathcal{M} = \{\mathbf{x} \in \mathbb{R}^3, \mathbf{a}^\top \mathbf{x} + \alpha > 0\}$ and has IoR n, while the medium containing the object under scrutiny is located in $\overline{\mathcal{M}} = \{\mathbf{x} \in \mathbb{R}^3, \mathbf{a}^\top \mathbf{x} + \alpha \leq 0\}$ and has IoR $\overline{n}$, and we denote $\mu = n/\overline{n} < 1$. Lastly, for a plane $\mathcal{P}$ of equation $\mathbf{v}^\top \mathbf{x} + \beta = 0$ and $\mathbf{w} \in \mathbb{R}^3$ with $\mathbf{v}^\top \mathbf{w} \neq 0$, we define the projection on plane $\mathcal{P}$ along direction $\mathbf{w}$ as

$$\boldsymbol{P}_{\mathcal{P}}^{\mathbf{w}}(\mathbf{x}) = \mathbf{x} - \frac{\mathbf{v}^\top \mathbf{x} + \beta}{\mathbf{v}^\top \mathbf{w}} \mathbf{w} = \left(\mathrm{id} - \frac{\mathbf{w}\mathbf{v}^\top}{\mathbf{w}^\top \mathbf{v}} \right) \mathbf{x} - \frac{\beta \mathbf{w}}{\mathbf{v}^\top \mathbf{w}}. \tag{3.1}$$

The orthogonal projection on $\mathbf{v}^\top \mathbf{x} = 0$ is simply denoted by $\boldsymbol{P}_{\mathbf{v}^\perp}$.

Depth from the Interface. In the orthographic case, all the light rays reaching the camera are orthogonal to the image plane, i.e., parallel to $\mathbf{e}_3$. In the absence of a refractive interface (or when the interface is fronto-parallel as in [16,18]), the projection is simply $\mathbf{p}(\mathbf{x}) = \mathbf{\Pi x}$ and the surface parameterization, as its right inverse, is $\mathcal{S}(u,v) = (u,v,z(u,v))^\top$ with z the depth map. However, when a non fronto-parallel refractive interface comes into play, the projection becomes non-orthogonal (see Fig. 1b). In this case, the rays reaching the camera are parallel to direction $\mathbf{e}_3$, and come from parallel incident rays with common direction $\mathbf{r} \in \mathbb{S}^2$ within the refractive medium $\overline{\mathcal{M}}$ as the interface is planar. Vector $\mathbf{r}$ is fully determined by Snell's law (2.3) as it must refract to viewing direction $\mathbf{e}_3$:

$$\mathbf{r} = \text{Snell}_\mu^{-\mathbf{a}}(\mathbf{e}_3), \tag{3.2}$$

where the sign before $\mathbf{a}$ comes from the fact that $\mathbf{a}$ is oriented towards the camera, while $\mathbf{e}_3$ and $\mathbf{r}$ are oriented towards the surface (see Fig. 1b).

Therefore, a point $\mathbf{x} \in \overline{\mathcal{M}}$ on the immersed surface first projects non-orthogonally onto $\mathcal{D}$ along the refracted viewing direction $\mathbf{r}$, before being orthogonally projected onto the camera image plane along the viewing direction $\mathbf{e}_3$:

$$\mathbf{p}(\mathbf{x}) = \mathbf{\Pi} \boldsymbol{P}_{e_3^\perp}(\boldsymbol{P}_{\mathcal{D}}^{\mathbf{r}}(\mathbf{x})). \tag{3.3}$$

This leads to a straightforward model where we deproject pixel $(u,v)^\top$ in the image plane to a point on the refractive interface $\mathcal{D}$, and then follow the incident ray up to the object: $\mathcal{S}(u,v) = \boldsymbol{P}_{\mathcal{D}}^{e_3}(u,v,0)^\top + \bar{z}(u,v)\mathbf{r}$, with $\bar{z}$ the pseudo-depth (distance travelled along the refracted ray $\mathbf{r}$). One readily checks that $\mathbf{p}(\mathcal{S}(u,v)) = (u,v)^\top$. Using (3.1), we can write

$$\mathcal{S}(u,v) = \boldsymbol{P}_{\mathcal{D}}^{\mathbf{e}_3}(u,v,0)^\top + \bar{z}(u,v)\mathbf{r} = \mathbf{A}(u,v,0)^\top + \mathbf{t} + \bar{z}(u,v)\mathbf{r}, \tag{3.4}$$

with known quantities

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{a_1}{a_3} & -\frac{a_2}{a_3} & 0 \end{pmatrix}, \quad \mathbf{t} = -\frac{\alpha}{a_3}\mathbf{e}_3. \tag{3.5}$$

Surface Normals. Now, let us establish the link between the pseudo-depth $\bar{z}$ from the interface, and the normal $\mathbf{n}$ to the surface. To do this, let us consider the partial derivatives of the parameterization. They are given by $\mathcal{S}_u = \mathbf{A}\mathbf{e}_1 + \bar{z}_u\mathbf{r}$ and $\mathcal{S}_v = \mathbf{A}\mathbf{e}_2 + \bar{z}_v\mathbf{r}$. An (unnormalized) normal to the surface $\mathcal{S}(u,v)$ is $\mathcal{S}_u \times \mathcal{S}_v = (\mathbf{A}\mathbf{e}_1 + \bar{z}_u\mathbf{r}) \times (\mathbf{A}\mathbf{e}_2 + \bar{z}_v\mathbf{r})$. Set $\mathbf{b}^1 = \mathbf{r} \times \mathbf{A}\mathbf{e}_2$, $\mathbf{b}^2 = \mathbf{A}\mathbf{e}_1 \times \mathbf{r}$ and $\mathbf{b}^3 = \mathbf{A}\mathbf{e}_2 \times \mathbf{A}\mathbf{e}_1$. Then $\mathcal{S}_u \times \mathcal{S}_v = \bar{z}_u\mathbf{b}^1 + \bar{z}_v\mathbf{b}^2 - \mathbf{b}^3$. By letting $\mathbf{B}$ be the matrix $-(\mathbf{b}^1, \mathbf{b}^2, \mathbf{b}^3)$,

$$\mathbf{n}(u,v) = \mathbf{n}(\mathcal{S}(u,v)) \propto \mathbf{B}\begin{pmatrix} \nabla\bar{z}(u,v) \\ -1 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} \frac{a_2 r_2}{a_3} + r_3 & -\frac{a_1 r_2}{a_3} & \frac{a_1}{a_3} \\ -\frac{a_2 r_1}{a_3} & \frac{a_1 r_1}{a_3} + r_3 & \frac{a_2}{a_3} \\ -r_1 & -r_2 & 1 \end{pmatrix}. \tag{3.6}$$

Equation (3.6) relates the surface normals to the underlying gradient of the pseudo-depth from the interface. When the interface is fronto-parallel, $\mathbf{a} = -\mathbf{e}_3$, $\mathbf{r} = \mathbf{e}_3$ and $\bar{z} = z - \beta$. Hence, $\mathbf{B} = \mathbf{I}_3$ and the formula matches the classic one obtained in the absence of refraction: $\mathbf{n}(u,v) \propto \left(\nabla z(u,v)^\top, -1\right)^\top$.

4 Image Formation Model Under Directional Lighting

Now, we turn our attention to extending the image formation model (2.1) to the refractive case. We assume to have at hand a series of k images $I_1, \dots, I_k$, with lighting directions $\mathbf{s}_1, \dots, \mathbf{s}_k$ and densities $\varphi_1, \dots, \varphi_k$ calibrated inside $\mathcal{M}$. The effective lighting inside $\overline{\mathcal{M}}$ will however be different from the calibrated one, due to refraction.

Effective Lighting Directions and Densities. We assume that light directions and densities are known in the camera medium $\mathcal{M}$ thanks to calibration, and we denote these calibrated parameters by $\mathbf{s}_i$ and φ_i. However, the directions $\overline{\mathbf{s}}_i$ and densities $\overline{\varphi}_i$ of the effective light beams reaching the surface differ from calibrated values, see Fig. 2.

After crossing the refractive interface, the incident light beams obviously remain parallel, yet their directions become, according to Snell's law (2.3):

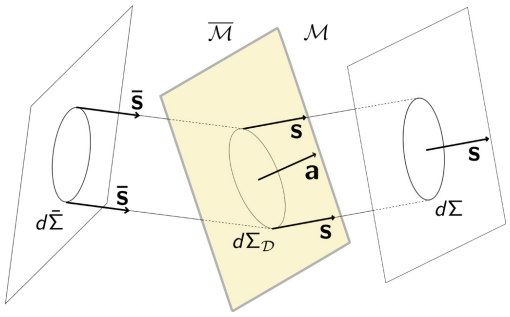


Fig. 2. Light refraction by a planar interface with normal $\mathbf{a}$, with $\mathbf{s}$ the light direction calibrated outside the refractive medium, and $\overline{\mathbf{s}}$ the effective refracted direction (in this drawing, the light source is on the right). Light direction is changed according to Snell's law, while its density is multiplied by $\frac{\mathbf{a}^\top \mathbf{s}}{\mathbf{a}^\top \overline{\mathbf{s}}}$.

$$\overline{\mathbf{s}}_i = -\mathrm{Snell}_{\mu}^{-\mathbf{a}}(-\mathbf{s}_i), \quad i \in \{1, \dots, k\}. \tag{4.1}$$

Moreover, the size of the surface elements orthogonal to the rays also changes, according to $\frac{d\overline{\Sigma}_i}{\mathbf{a}^\top \overline{\mathbf{s}}_i} = d\Sigma_{\mathcal{D}} = \frac{d\Sigma_i}{\mathbf{a}^\top \mathbf{s}_i}$. Then:

$$\overline{\varphi}_i = \frac{\mathbf{a}^\top \mathbf{s}_i}{\mathbf{a}^\top \overline{\mathbf{s}}_i}\, \varphi_i, \quad i \in \{1, \dots, k\}. \tag{4.2}$$

Let us emphasize that, even if all the sources have exactly the same intensity i.e., $\varphi_i = \varphi_j, \forall i \neq j$, the effective densities will be different. For instance, when $n = 1$, $\overline{n} = 1.5$, and $\varphi_1 = \varphi_2 = 1$, a lighting orthogonal to the interface yields $\overline{\varphi}_1 = 1$, while an incident angle of 30° yields $\overline{\varphi}_2 = 0.91$. This effect is thus far from negligible in a calibrated PS setup.

Fresnel Coefficients. The interface may act partially as a mirror, with the amount of transmitted light being a function of the incident angle. This happens twice in the process: first when going from the light source in $\mathcal{M}$ to the surface embedded in $\overline{\mathcal{M}}$, and then when going from the latter to the camera, back in $\mathcal{M}$.

The incident and outgoing angles when going from $\mathcal{M}$ towards $\overline{\mathcal{M}}$ will vary depending on the incident direction $\mathbf{s}_i$, $i \in \{1, \dots, k\}$: each light source will thus induce a different transmission rate. This rate is however the same whatever the

point $\mathbf{x}$, since lighting is assumed directional - this would not be the case for instance under a near point light source. Assuming that all the light beams are unpolarized, each transmission rate is given by the Fresnel coefficient

$$T_i^{\mathcal{M}\to\overline{\mathcal{M}}} = 1 - \frac{1}{2}\left(\frac{\left(\mu\,\mathbf{a}^\top\mathbf{s}_i - \mathbf{a}^\top\overline{\mathbf{s}}_i\right)^2}{\left(\mu\,\mathbf{a}^\top\mathbf{s}_i + \mathbf{a}^\top\overline{\mathbf{s}}_i\right)^2} + \frac{\left(\mu\,\mathbf{a}^\top\overline{\mathbf{s}}_i - \mathbf{a}^\top\mathbf{s}_i\right)^2}{\left(\mu\,\mathbf{a}^\top\overline{\mathbf{s}}_i + \mathbf{a}^\top\mathbf{s}_i\right)^2}\right),\ i \in \{1,\dots,k\}. \tag{4.3}$$

Taking again as an example the case $n = 1$, $\overline{n} = 1.5$, an incident lighting orthogonal to the interface yields $T_1^{\mathcal{M}\to\overline{\mathcal{M}}} = 0.9600$, while an incident angle of 30° yields $T_2^{\mathcal{M}\to\overline{\mathcal{M}}} = 0.9585$. This shows that this Fresnel coefficient is non-negligible, although less dramatic than the change in the incident densities.

When going from $\overline{\mathcal{M}}$ to $\mathcal{M}$, the incident and outgoing angles are the same for all images $I_1,\dots,I_k$ (viewing direction is independent from the incident lighting directions), therefore the transmission rate simply scales all the brightness values at pixel $\mathbf{p}$ conjugate to $\mathbf{x}$ by the same coefficient $T^{\overline{\mathcal{M}}\to\mathcal{M}}(\mathbf{x})$, $\forall i \in \{1,\dots,k\}$. Besides, since we assume orthographic viewing, these angles are the same for all pixels, hence $T^{\overline{\mathcal{M}}\to\mathcal{M}}$ is independent from $\mathbf{x}$ as well - this would not be the case under pinhole projection. The Fresnel coefficient is then written as

$$T^{\overline{\mathcal{M}}\to\mathcal{M}} = 1 - \frac{1}{2}\left(\frac{\left(-\mathbf{a}^\top\mathbf{r} + \mu\,\mathbf{a}^\top\mathbf{e}_3\right)^2}{\left(-\mathbf{a}^\top\mathbf{r} - \mu\,\mathbf{a}^\top\mathbf{e}_3\right)^2} + \frac{\left(-\mathbf{a}^\top\mathbf{e}_3 + \mu\,\mathbf{a}^\top\mathbf{r}\right)^2}{\left(-\mathbf{a}^\top\mathbf{e}_3 - \mu\,\mathbf{a}^\top\mathbf{r}\right)^2}\right). \tag{4.4}$$

Note that this second Fresnel coefficient simply scales all the observations by the same constant, hence it can be taken into account by normalization.

Forward Model for Refractive PS. To summarize the effects described above, in the presence of refraction the classic image formation model (2.1) becomes

$$I_i(\mathbf{p}) = \underbrace{\left(\overline{\varphi}_i T_i^{\mathcal{M}\to\overline{\mathcal{M}}}\right)}_{:=\overline{\psi}_i}\ \underbrace{\left(T^{\overline{\mathcal{M}}\to\mathcal{M}}\rho(\mathbf{x})\right)}_{:=\varrho(\mathbf{x})}\ \mathbf{n}(\mathbf{x})^\top\ \underbrace{\left(-\mathrm{Snell}_\mu^{-\mathbf{a}}(-\mathbf{s}_i)\right)}_{:=\overline{\mathbf{s}}_i},\ i \in \{1,\dots,k\}, \tag{4.5}$$

where:

- the effective lighting directions $\overline{\mathbf{s}}_i$ must be deduced from the calibrated ones $\mathbf{s}_i$ according to Snell's law (4.1);
- the effective lighting densities $\overline{\psi}_i$ must be deduced from the calibrated ones φ_i using (4.2) (density attenuation) and (4.3) (Fresnel coefficients);
- the Fresnel-scaled albedo $\varrho(\mathbf{x})$ (see (4.4)) and the surface normal $\mathbf{n}(\mathbf{x})$ (see (3.6)) constitute the unknowns of the PS problem.

To summarize, we have established the geometric parameterization of the surface, and shown how to deduce the effective lighting directions and densities from the ones calibrated outside the refractive medium. In the next section, we turn our attention to the numerical resolution of the system of Eqs. (4.5), by adapting state-of-the-art strategies.

5 Solving Refractive PS

To invert the image formation model (4.5), it is possible to either sequentially estimate normals and the 3D-shape, or to directly compute the 3D-shape.

Normal and Albedo Estimation. Estimating the surface normals and albedo comes down to solving the system of Eqs. (4.5) with known effective lighting densities $\overline{\psi}_i$ and effective incident lighting directions $\bar{\mathbf{s}}_i$. This system of equations admits a unique approximate solution as long as $k \geq 3$ and the effective directions $\bar{\mathbf{s}}_i$ are non-coplanar (which is the case if the incident directions $\mathbf{s}_i$ are themselves non-coplanar). Any calibrated PS method can be applied for this task, simply changing the light directions and densities so as to take refraction into account. For instance, defining $\mathbf{m} := \varrho\,\mathbf{n}$, one may consider the following pixelwise linear least-squares solution, $\forall \mathbf{p}$:

$$\mathbf{m}(\mathbf{p}) = \underset{\mathbf{m}\in\mathbb{R}^3}{\operatorname{argmin}} \sum_{i=1}^{k} \left(\overline{\psi}_i \bar{\mathbf{s}}_i^\top \mathbf{m} - I_i(\mathbf{p})\right)^2, \;\; \varrho(\mathbf{x}) = |\mathbf{m}(\mathbf{p})|, \;\; \mathbf{n}(\mathbf{x}) = \frac{\mathbf{m}(\mathbf{p})}{|\mathbf{m}(\mathbf{p})|}, \quad (5.1)$$

which can be computed in closed-form by using the pseudo-inverse. If robustness (e.g., to shadows or specularities) needs to be addressed, more evolved solutions based on deep neural networks [7] can be considered. Semi-calibrated algorithms [8] could also be employed for automatically inferring the coefficients $\overline{\psi}_i$. Provided that the integrability constraint [28] is adapted to the refractive case, uncalibrated algorithms [12] would even provide the $\bar{\mathbf{s}}_i$ up to a generalized bas-relief ambiguity [2], which could be resolved a posteriori using one of the methods discussed in [24].

Normal Integration. The next stage consists in obtaining the surface from its normals. Equation (3.6) tells us that once $\mathbf{n}(u, v)$ is estimated, computing $\mathbf{B}^{-1}\mathbf{n}(u, v)$ using the definition in (3.6) of $\mathbf{B}$, and then normalizing both its first components by the third one provides an estimate for $\nabla\bar{z}(u, v)$. Given these gradient estimates, the pseudo-depth map from the interface can be obtained by integration. Any approach designed for the classic case can be employed at this stage, just changing the input gradient estimates (see [20]). Once the pseudo-depth has been computed, one simply has to apply Eq. (3.4) to obtain the 3D-surface.

Direct Differential Approach. To avoid bias accumulation due to the sequential estimation of normals and shape, it is also possible to follow a direct differential approach. Plugging (3.6) into (4.5), we get, $\forall (i, \mathbf{p})$:

$$I^i(\mathbf{p}) = \overline{\psi}_i \underbrace{\frac{\varrho(\mathbf{x})}{\left|\mathbf{B}\begin{pmatrix}\nabla\bar{z}(\mathbf{p})\\ -1\end{pmatrix}\right|}}_{:=\tilde{\varrho}(\mathbf{p})} \underbrace{\left(\mathbf{B}^\top\bar{\mathbf{s}}_i\right)^\top}_{:=\tilde{\mathbf{s}}_i^\top} \begin{pmatrix}\nabla\bar{z}(\mathbf{p})\\ -1\end{pmatrix}, \quad (5.2)$$

which is a system of nonlinear PDEs. Therein, $\tilde{\varrho}$ will be considered as the unknown "pseudo-albedo" and vectors $\tilde{\mathbf{s}}_i$ as known "pseudo light vectors". The direct joint estimation of the pseudo-albedo and the pseudo-depth from the interface can then be written as a variational problem:

$$\min_{\bar{z},\tilde{\varrho}} \sum_{\mathbf{p}} \sum_{i} \Phi \left(\overline{\psi}_i \tilde{\varrho}(\mathbf{p}) \tilde{\mathbf{s}}_i^{\top} \begin{pmatrix} \nabla \bar{z}(\mathbf{p}) \\ -1 \end{pmatrix} - I_i(\mathbf{p}) \right), \tag{5.3}$$

using some robust estimator Φ and a finite differences approximation of the gradient operator. Once the depth from the interface and the pseudo-albedo have been estimated, it only remains to deduce the true Fresnel-scaled albedo ϱ from $\tilde{\varrho}$ and $\nabla \bar{z}$ using the definition in Eq. (5.2), and eventually the 3D-surface by using Eq. (3.4). Again, such a differential approach can be extended to the semi-calibrated scenario [21], or even to refine the pseudo light vectors [22].

Validation on Synthetic and Real-World Data. In order to empirically validate our forward model and its inversion, we first generated synthetic PS images using Blender [1]. The Lambertian surfaces were placed inside glass ($\overline{n} = 1.5$) while the light sources and the orthographic camera were placed inside air ($n = 1$). In each experiment, 12 images were rendered under varying parallel lighting, whose direction and relative density are provided by the engine.

We first considered images of a perfect sphere. We used the sequential approach (5.1) followed by DCT integration [20], as well as the differential approach (5.3) with Cauchy estimator [22]. In both cases, we carried out 3D-reconstruction first neglecting all refraction effects, and then with refraction considered. To quantitatively evaluate the results, we fit a sphere to the 3D-reconstruction using least-squares, and compute the normalized RMSE between the 3D-reconstruction and the spherical fit. Results are shown in Table 1. It can be seen that for both approaches, considering refraction drastically improves the 3D-reconstruction, even when the interface is not rotated. Indeed, as can be seen in Fig. 3, neglecting refraction causes the 3D-reconstruction to "flatten".

Table 1. Normalized root mean square error between the estimated surface and a least-squares spherical fit, for a planar refractive interface (the angles stand for the rotations around the horizontal and vertical axes, respectively). Considering refraction systematically improves performance, for both the sequential and the differential approaches.

	No interface	$(0°, 0°)$	$(11.5°, 0°)$	$(11.5°, 22.5°)$
Sequential w/o refraction	0.0035	0.0195	0.0232	0.0403
Sequential w/ refraction	0.0035	0.0116	0.0129	0.0261
Differential w/o refraction	0.0020	0.0202	0.0239	0.0396
Differential w/ refraction	**0.0020**	**0.0114**	**0.0127**	**0.0254**

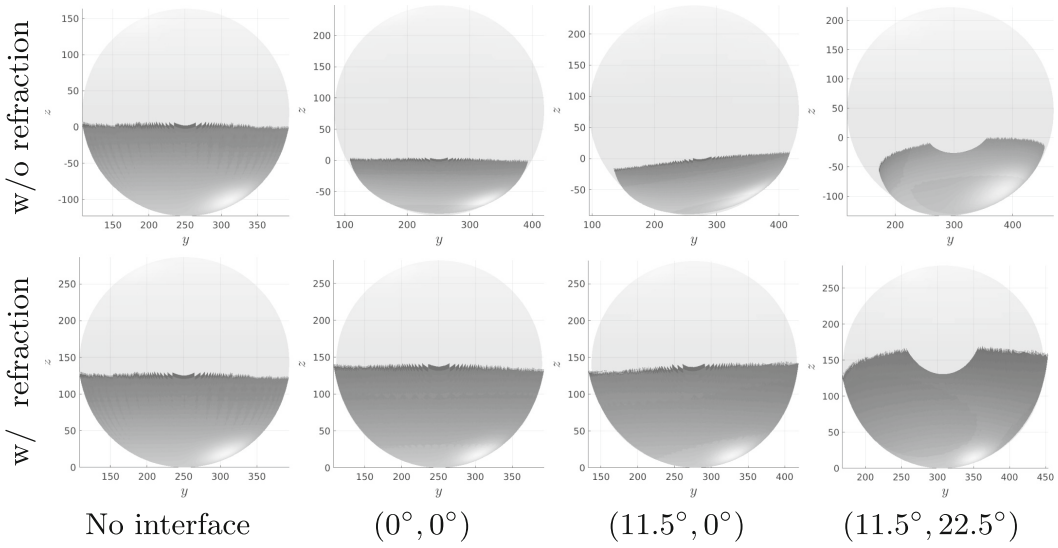


Fig. 3. 3D-reconstructions of the spheres from Table 1 using the differential approach, neglecting (top) or considering (bottom) refraction. The light grey spheres are the least-squares spherical fits to the estimated surfaces used for the quantitative evaluation in Table 1. Neglecting refraction induces a severe "flattening".

Then, we replaced the sphere by two objects with a more complex shape: an insect (imaged with the interface rotated by 5° around the horizontal axis) and a skull (imaged with the interface rotated by 20° around the horizontal axis, and by 11.25° around the vertical one). The results in Fig. 4, obtained with the differential approach, show that it is possible to achieve a 3D-reconstruction which is indistinguishable from the one obtained in the absence of refraction. In particular, the "flattening" effect is corrected.

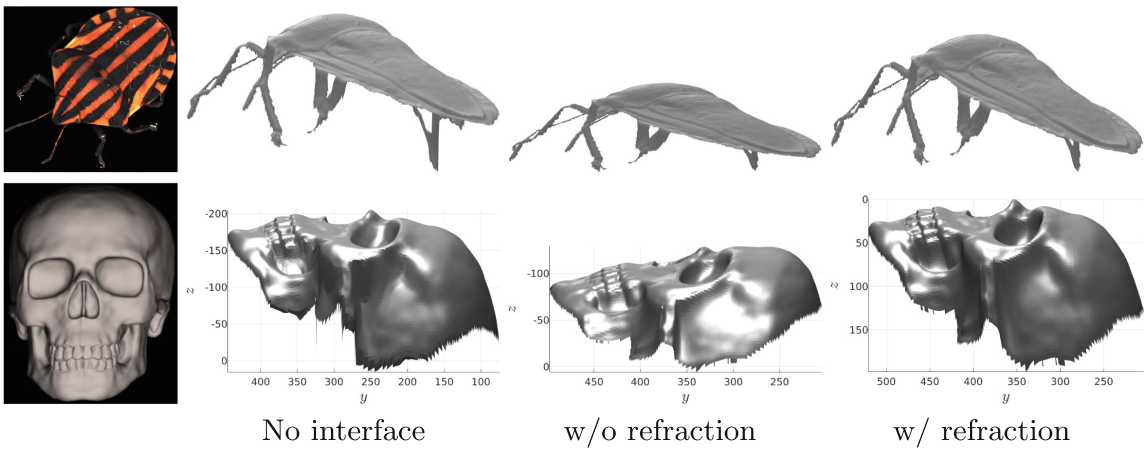


Fig. 4. 3D-reconstruction of an insect (top) and a skull (bottom). In each row, the first image represents one of the input images (out of 12); the second one shows the 3D-reconstruction obtained in the absence of the interface (for reference); and the other ones show the 3D-reconstruction in the presence of the interface, while neglecting or considering refraction.

Lastly, we conducted experiments on a real-world dataset. Our acquisition setup, illustrated in Fig. 1a, consists of 8 calibrated directional light sources. A diffuse sphere was imaged in the air, and then immersed in an aquarium filled with pure water (see Fig. 5). We performed the 3D-reconstruction using (5.3), and compared the results neglecting or considering refraction effects. For both a fronto-parallel and a rotated interface, considering refraction largely reduces the flattening and distortion effects, which empirically validates our method.

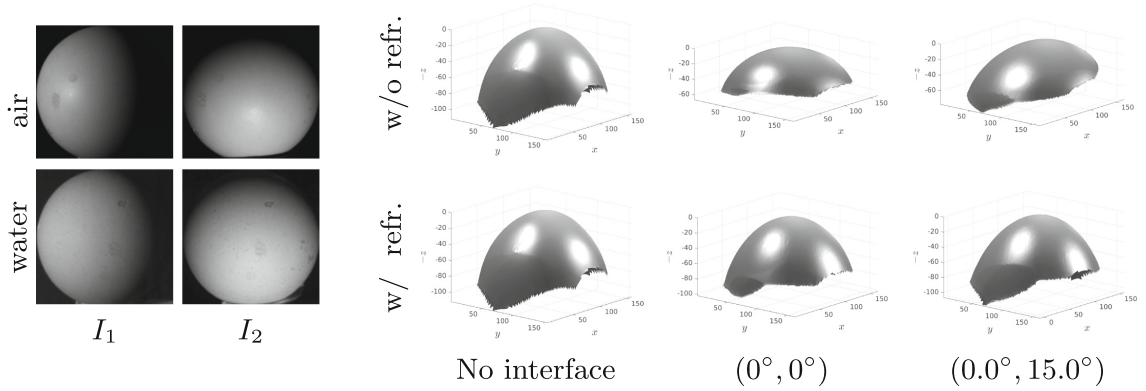


Fig. 5. 3D-reconstruction of a real-world sphere. On the left, we show two of the input images, in the absence ("air") and in presence ("water") of a refractive interface which is rotated by 15.0° around the vertical axis. On the right, we show the 3D-reconstruction of the sphere, taking into account (top) or not (bottom) refraction effects, in three cases: in the air, with a fronto-parallel interface, and with a rotated interface. Neglecting refraction leads to flattened and distorded 3D-reconstructions, while these effects are much attenuated with the proposed approach.

6 Conclusion and Future Work

In this paper, we have explored the impact of the presence of a refractive interface on the modeling of the photometric stereo problem, both in terms of geometry and of photometric image formation model. We further showed how to adapt existing solutions so as to take into account geometric deformation, refraction of incident directions, attenuation of densities and Fresnel coefficients. We showed that taking into account such phenomena largely improves the accuracy of the 3D-reconstruction. However, the explicit modeling of refraction effects was eased by a few simplifying assumptions: orthographic viewing, directional lighting, planar interface and absence of light absorption. In the future, we plan to explore the changes induced by the relaxation of these assumptions. This can partially be achieved by making the forward more realistic through the incorporation of, e.g., a refractive near-field illumination model [23] or distance-dependent light attenuation [16,26]. However, we believe that differentiable inverse rendering frameworks may constitute an even more promising track for solving nonstandard photometric 3D-reconstruction problems in a somehow generic manner.

Such approaches have recently been successfully employed for solving complex multi-view 3D-reconstruction problems [15], yet for now they remain limited to cases where the surface projection onto the camera comes down to a simple rasterization. To cope with evolved refractive effects, one could thus imagine combining differentiable inverse rendering with powerful renderers such as Mitsuba 2 [19].

Acknowledgements. This project was partially supported by the KU Data+ Project Phylorama, the ALICIA-Vision LabCom (ANR-19-LCV1-0002), and the Inclusive Museum Guide project (ANR-20-CE38-0007).

References

1. Blender - a 3D modelling and rendering package. http://www.blender.org
2. Belhumeur, P.N., Kriegman, D.J., Yuille, A.L.: The bas-relief ambiguity. IJCV **35**(1), 33–44 (1999)
3. Cassidy, M., Mélou, J., Quéau, Y., Lauze, F., Durou, J.D.: Refractive multi-view stereo. In: 3DV (2020)
4. Chadebecq, F., et al.: Refractive two-view reconstruction for underwater 3D vision. IJCV **128**(5), 1101–1117 (2020)
5. Chari, V., Sturm, P.: Multiple-view geometry of the refractive plane. In: BMVC (2009)
6. Chen, C., Wang, H., Zhang, Z., Gao, F.: Three-dimensional reconstruction from a fringe projection system through a planar transparent medium. OptEx **30**(19), 34824–34834 (2022)
7. Chen, G., Han, K., Shi, B., Matsushita, Y., Wong, K.Y.K.: Deep photometric stereo for non-Lambertian surfaces. PAMI **44**(1), 129–142 (2020)
8. Cho, D., Matsushita, Y., Tai, Y.W., Kweon, I.S.: Semi-calibrated photometric stereo. PAMI **42**(1), 232–245 (2018)
9. Fan, H., et al.: Underwater optical 3-D reconstruction of photometric stereo considering light refraction and attenuation. IEEE J. Ocean. Eng. **47**(1), 46–58 (2021)
10. Fujimura, Y., Iiyama, M., Hashimoto, A., Minoh, M.: Photometric stereo in participating media considering shape-dependent forward scatter. In: CVPR (2018)
11. Fujitomi, T., Sakurada, K., Hamaguchi, R., Shishido, H., Onishi, M., Kameda, Y.: LB-NERF: light bending neural radiance fields for transparent medium. In: ICIP (2022)
12. Hayakawa, H.: Photometric stereo under a light source with arbitrary motion. JOSA A **11**(11), 3079–3089 (1994)
13. Hu, X., Lauze, F., Pedersen, K.S.: Refractive pose refinement. IJCV (2023). https://doi.org/10.1007/s11263-023-01763-4
14. Mikš, A., Novák, P.: Determination of unit normal vectors of aspherical surfaces given unit directional vectors of incoming and outgoing rays: comment. JOSA A **29**(7), 1356–1357 (2012)
15. Munkberg, J., et al.: Extracting triangular 3D models, materials, and lighting from images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8280–8290 (2022)
16. Murai, S., Kuo, M.Y.J., Kawahara, R., Nobuhara, S., Nishino, K.: Surface normals and shape from water. In: CVPR (2019)

17. Murez, Z., Treibitz, T., Ramamoorthi, R., Kriegman, D.: Photometric stereo in a scattering medium. In: CVPR (2015)
18. Narasimhan, S.G., Nayar, S.K.: Structured light methods for underwater imaging: light stripe scanning and photometric stereo. In: OCEANS (2005)
19. Nimier-David, M., Vicini, D., Zeltner, T., Jakob, W.: Mitsuba 2: a retargetable forward and inverse renderer. ACM Trans. Graph. (TOG) **38**(6), 1–17 (2019)
20. Quéau, Y., Durou, J.D., Aujol, J.F.: Normal integration: a survey. JMIV **60**(4), 576–593 (2018)
21. Quéau, Y., Wu, T., Cremers, D.: Semi-calibrated near-light photometric stereo. In: Lauze, F., Dong, Y., Dahl, A.B. (eds.) SSVM 2017. LNCS, vol. 10302, pp. 656–668. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58771-4_52
22. Quéau, Y., Wu, T., Lauze, F., Durou, J.D., Cremers, D.: A non-convex variational approach to photometric stereo under inaccurate lighting. In: CVPR (2017)
23. Sanao, H., Yingjie, S., Ming, L., Jingwei, Q., Ke, X.: Underwater 3D reconstruction using a photometric stereo with illuminance estimation. Appl. Opt. **62**(3), 612–619 (2023)
24. Shi, B., Wu, Z., Mo, Z., Duan, D., Yeung, S.K., Tan, P.: A benchmark dataset and evaluation for non-Lambertian and uncalibrated photometric stereo. In: CVPR (2016)
25. Tsiotsios, C., Angelopoulou, M.E., Kim, T.K., Davison, A.J.: Backscatter compensated photometric stereo with 3 sources. In: CVPR (2014)
26. Tsiotsios, C., Davison, A.J., Kim, T.K.: Near-lighting photometric stereo for unknown scene distance and medium attenuation. IVC **57**, 44–57 (2017)
27. Woodham, R.J.: Photometric stereo: a reflectance map technique for determining surface orientation from image intensity. In: Image understanding systems and industrial applications I, vol. 155, pp. 136–143 (1979)
28. Yuille, A.L., Snow, D., Epstein, R., Belhumeur, P.N.: Determining generative models of objects under varying illumination: shape and albedo from multiple images using SVD and integrability. IJCV **35**(3), 203–222 (1999)

Chapter D

Fourth paper

RNb-NeuS: Reflectance and Normal-based Multi-View 3D Reconstruction

Baptiste Brument[1,*] Robin Bruneau[1,2,*] Yvain Quéau[3] Jean Mélou[1]
François Bernard Lauze[2] Jean-Denis Durou[1] Lilian Calvet[4]

[1]IRIT, UMR CNRS 5505, Toulouse, France
[2]DIKU, Copenhagen, Denmark
[3]Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, Caen, France
[4]OR-X, Balgrist Hospital, University of Zurich, Zürich, Switzerland

Abstract

This paper introduces a versatile paradigm for integrating multi-view reflectance (optional) and normal maps acquired through photometric stereo. Our approach employs a pixel-wise joint re-parameterization of reflectance and normal, considering them as a vector of radiances rendered under simulated, varying illumination. This re-parameterization enables the seamless integration of reflectance and normal maps as input data in neural volume rendering-based 3D reconstruction while preserving a single optimization objective. In contrast, recent multi-view photometric stereo (MVPS) methods depend on multiple, potentially conflicting objectives. Despite its apparent simplicity, our proposed approach outperforms state-of-the-art approaches in MVPS benchmarks across F-score, Chamfer distance, and mean angular error metrics. Notably, it significantly improves the detailed 3D reconstruction of areas with high curvature or low visibility.

1. Introduction

Automatic 3D reconstruction is pivotal in various fields, such as archaeological and cultural heritage (virtual reconstruction), medical imaging (surgical planning), virtual and augmented reality, games and film production.

Multi-view stereo (MVS) [5], which retrieves the geometry of a scene seen from multiple viewpoints, is the most famous 3D reconstruction solution. Coupled with neural volumetric rendering (NVR) techniques [22], it effectively handles complex structures and self-occlusions. However, dealing with non-Lambertian scenes remains a challenge due to the breakdown of the underlying brightness consistency assumption. The problem is also ill-posed in certain configurations e.g., poorly textured scene [25] or degenerate viewpoints configurations with limited baselines. Moreover, despite recent efforts in this direction [13], recovering the thinnest geometric details remains difficult under fixed illumination. In such a setting, estimating the reflectance of the scene also remains a challenge.

On the other hand, photometric stereo (PS) [24], which relies on a collection of images acquired under varying lighting, excels in the recovery of high-frequency details under the form of normal maps. It is also the only photographic technique that can estimate reflectance. And, with the recent advent of deep learning techniques [8], PS gained enough maturity to handle non-Lambertian surfaces and complex illumination. Yet, its reconstruction of geometry's low frequencies remains suboptimal.

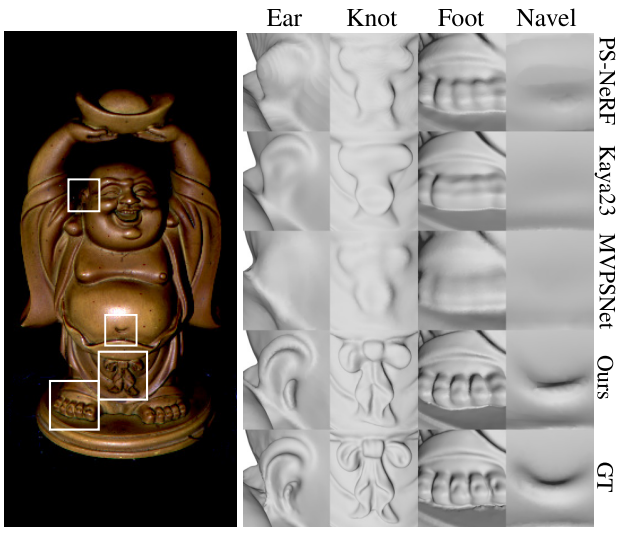


Figure 1. One image from DiLiGenT-MV's Buddha dataset [12], and 3D reconstruction results from several recent MVPS methods: [11, 26, 27] and ours. The latter provides the fine details closest to the ground truth (GT), while being remarkably simpler.

*Equal contributions. brument.bcb@gmail.com / rb@di.ku.dk

Given these complementary characteristics, the integration of MVS and PS seems natural. This integration, known as multi-view photometric stereo (MVPS), aims to reconstruct geometry from multiple views and illumination conditions. Recent MVPS solutions jointly solve MVS and PS within a multi-objective optimization, potentially losing the thinnest details due to the possible incompatibility of these objectives – see Fig. 1. In this work, we explore a simpler route for solving MVPS by decoupling the two problems.

We start with the observation that recent PS techniques deliver exceptionally high-quality reflectance and normal maps, which we use as input data. To accurately reconstruct the surface reflectance and geometry, we need to fuse these maps, a challenging task within a single-objective optimization due to their inhomogeneity. Our method provides a solution to this problem by combining NVR with a simple and effective pixel-wise re-parameterization.

In this method, the input reflectance and normal for each pixel are merged into a vector of radiances simulated under arbitrary, varying illumination. We then adapt an NVR pipeline to optimize the consistency of these simulations wrt to the scene reflectance and geometry, modeled as the zero-level set of a trained signed distance function (SDF). Coupled with a state-of-the-art PS method such as [8] for obtaining the input reflectance and normals, this approach yields an MVPS pipeline reaching an unprecedented level of fine details, as illustrated in Fig. 1. Besides being the first to exploit reflectance as a prior, our proposed MVPS paradigm is extremely versatile, compatible with any existing or future PS method, whether calibrated or uncalibrated, deep learning-based, or classic optimization procedures.

The rest of this work is organized as follows. Sect. 2 discusses state-of-the-art MVPS methods. The proposed 3D reconstruction from reflectance and normals is detailed in Sect. 3. Sect. 4 then sketches a proposal for an MVPS algorithm based on this approach. Sect. 5 extensively evaluates this algorithm, before our conclusions are drawn in Sect. 6.

2. Related work

Classical methods The first paper to deal with MVPS is by Hernandez et al. [6]. To avoid having to arbitrate the conflicts between the different normal maps, a 3D mesh is iteratively deformed, starting from the visual hull until the images recomputed using the Lambertian model match the original images, while penalizing the discrepancy between the PS normals and those of the 3D mesh. No prior knowledge of camera poses or illumination is required. Under the same assumptions, Park et al. [19, 20] start from a 3D mesh obtained by SfM (structure-from-motion) and MVS. Simultaneous estimation of reflectance, normals and illumination is achieved by uncalibrated PS, using the normals from the 3D mesh to remove the ambiguity, and estimating the details of the relief through 2D displacement maps.

MVPS is solved for the first time with a SDF representation of the surface by Logothetis et al. [14]. Therein, illumination is represented as near point light sources which are assumed calibrated, as well as the camera poses. Thanks to a voxel-based implementation, the surface details are better rendered than with the method of Park et al. [20].

Li et al [12] refine a 3D mesh obtained by propagating the SfM points according to [17], and estimate the BRDF using a calibrated setup. The creation of the public dataset "DiLiGenT-MV" validates numerically the improved results, in comparison with those of [20].

Deep learning-based methods Kaya et al. [10] proposed a solution to MVPS based on neural radiance fields (NeRFs) [16]. For each viewpoint, a normal map is obtained using a pre-trained PS network, before a NeRF is adapted to account for input surface normals from PS in the color function. The recovered geometry yet remains perfectible, according to [9]. Therein, the authors propose learning an SDF function whose zero level set best explains pixel depth and normal maps obtained by a pre-trained MVS [21] or PS network [7], respectively. To manage conflicting objectives in the proposed multi-objective optimization and get the best out of MVS and PS predictions, both networks are modified to output uncertainty measures on depth and normal predictions. The SDF optimization is then carried out while accounting for the inferred uncertainties.

PS-NeRF [26] solves MVPS by jointly estimating the geometry, material and illumination. To this end, the authors propose to regularize the gradient of a UNISURF [18] using the normal maps from PS, while relying on multi-layer perceptrons (MLPs) to explicitly model surface normals, BRDF, illumination, and visibility. These MLPs are optimized based on a shadow-aware differentiable rendering layer. A similar track is followed in [2], where NeRFs are combined with a physically-based differentiable renderer.

Such NeRF-based approaches provide undeniably better 3D reconstructions than classical methods, yet they remain computationally intensive. Recently, Zhao et al. [27] proposed a fast deep learning-based solution to MVPS. Aggregated shading patterns are matched across viewpoints so that to predict pixel depths and normal maps.

In [11], the authors proposed to complement the solution of [9] by adding a NVR loss term in order to benefit from the reliability of NVR in reconstructing objects with diverse material types. However, this results in a multi-objective optimization comprising three loss terms (besides the Eikonal term). However, similar to [9], the uncertainty-based hyper-parameter tuning does not completely eliminate conflicting objectives, which may induce a loss of fine-scale details. In contrast, we propose a single objective optimization based on an ad hoc re-parametrization which leads to the seamless integration of PS results in standard NVR pipelines. This is detailed in the next paragraph.

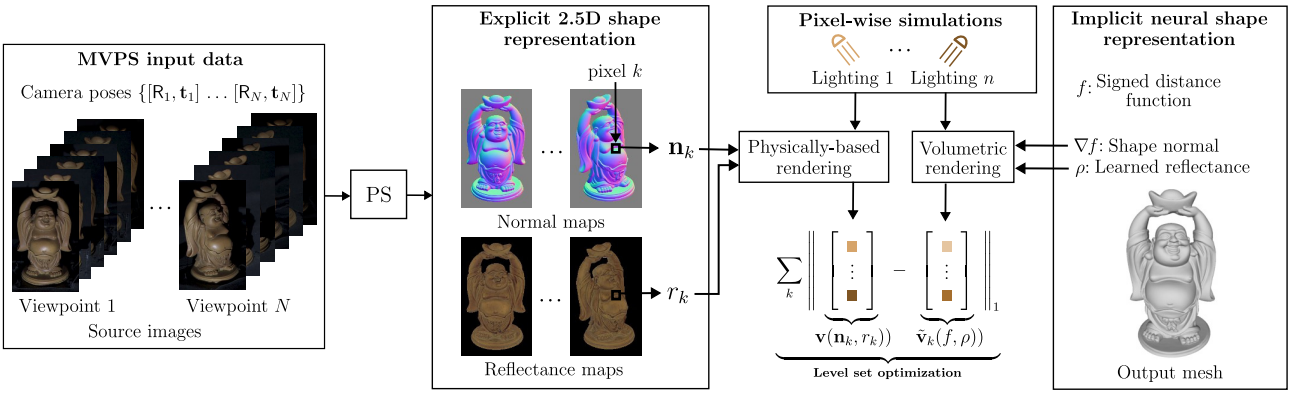


Figure 2. Overview of the proposed MVPS pipeline. The reflectance and normal maps provided for each view by PS are fused, by combining volume rendering with a pixel-wise re-parameterization of the inputs using physically-based rendering.

3. Proposed approach

Our aim is to infer a surface whose geometric and photometric properties are consistent with the per-view PS results. To do so, we resort to a volume rendering framework coupled with a re-parameterization of the inputs, as illustrated in Fig. 2 and detailed in the rest of this section.

3.1. Overview

Input data From the N image sets captured under fixed viewpoint and varying illumination, PS provides N reflectance and normal maps, out of which we extract a batch of m posed reflectance and normal values $\{r_k \in \mathbb{R}, \mathbf{n}_k \in \mathbb{S}^2\}_{k=1\dots m}$. Here, the normal vectors are expressed in world coordinates using the known camera poses. The input reflectance is without loss of generality represented by a scalar (albedo). Let us emphasize that this assumption does not imply that the observed scene must be Lambertian, but rather that we use only the diffuse component of the estimated reflectance. Using other reflectance components (specularity, roughness, etc.), if available, would represent a straightforward extension to more evolved physically-based rendering (PBR) models. Yet, we leave such an extension to perspective for now, since there are few PS methods reliably providing such data. Also, if the PS method provides no reflectance, one can set $r_k \equiv 1$ and use the proposed framework for multi-view normal integration.

Surface parameterization Our aim is to infer a 3D model of a scene, which consists of both a geometric map $f : \mathbb{R}^3 \to \mathbb{R}$ and a photometric one $\rho : \mathbb{R}^3 \to \mathbb{R}$. Therein, f associates a 3D point with its signed distance to the surface, which is thus given by the zero level set of f: $\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 \,|\, f(\mathbf{x}) = 0\}$. Regarding ρ, it encodes the reflectance associated with a 3D point. For input consistency, ρ is considered as a scalar function (albedo), though more advanced PBR models could again be incorporated.

Objective function Our method builds upon a re-parameterization $\mathbf{v} : \mathbb{S}^2 \times \mathbb{R} \to \mathbb{R}^n$ which combines a surface normal $\mathbf{n}_k \in \mathbb{S}^2$ and a reflectance value $r_k \in \mathbb{R}$ into a vector $\mathbf{v}(\mathbf{n}_k, r_k) \in \mathbb{R}^n$ of n radiance values that are simulated by physically-based rendering, using an arbitrary image formation model under varying illumination. Given this re-parameterization, the 3D reconstruction problem amounts to minimizing the difference between a batch of m intensity vectors simulated either from the input data or from volume rendering with the same PBR model, along with a regularization on the SDF:

$$\min_{f,\rho} \sum_{k=1}^{m} \|\mathbf{v}(\mathbf{n}_k, r_k) - \tilde{\mathbf{v}}_k(f, \rho)\|_1 + \lambda \mathcal{L}_{\text{reg}}(f). \quad (1)$$

Here, $\{(\mathbf{n}_k, r_k)\}_{k=1\dots m}$ stands for the batch of input reflectance and normal values, $\mathbf{v}(\mathbf{n}_k, r_k)$ for the k-th intensity vector simulated from the input data, $\tilde{\mathbf{v}}_k(f, \rho)$ for the corresponding one simulated by volume rendering, and $\lambda > 0$ is a tunable hyper-parameter for balancing the data fidelity with the regularizer $\mathcal{L}_{\text{reg}}$. The actual optimization can then be carried out seamlessly by resorting to a volume rendering-based 3D reconstruction pipeline such as NeuS [22], given that both $\tilde{\mathbf{v}}_k(f, \rho)$ and $\mathbf{v}(\mathbf{n}_k, r_k)$ correspond to pixel intensities. Let us now detail how we simulate the latter intensities $\mathbf{v}(\mathbf{n}_k, r_k)$ from the input reflectance and normal data.

3.2. Reflectance and normal re-parameterization

The input reflectance $\{r_k \in \mathbb{R}\}_k$ and normals $\{\mathbf{n}_k \in \mathbb{S}^2\}_k$ values constitute inhomogeneous quantities: the former are photometric scalars, and the latter geometric vectors lying on the three-dimensional unit sphere. Direct optimization of their consistency with the scene normal $\frac{\nabla f}{\|\nabla f\|}$ and albedo ρ would lead to multiple objectives balanced by hyper-parameters.

Instead, we propose to jointly re-parameterize the reflectance and normal data into a set of vectors $\{\mathbf{v}(\mathbf{n}_k, r_k) \in \mathbb{R}^n\}_k$ of homogeneous quantities, namely radiance values simulated using a PBR model under varying illumination. In order to enforce the bijectivity of this re-parameterization, we choose as PBR model the linear Lambertian one, under pixel-wise varying illumination represented by $n = 3$ arbitrary illumination vectors $\mathbf{l}_{k,1}, \mathbf{l}_{k,2}, \mathbf{l}_{k,3} \in \mathbb{R}^3$:

$$\begin{aligned}\mathbf{v}(\mathbf{n}_k, r_k) &= r_k[\mathbf{n}_k^\top \mathbf{l}_{k,1}, \mathbf{n}_k^\top \mathbf{l}_{k,2}, \mathbf{n}_k^\top \mathbf{l}_{k,3}]^\top \\ &= r_k \mathsf{L}_k \, \mathbf{n}_k,\end{aligned} \tag{2}$$

with $\mathsf{L}_k = [\mathbf{l}_{k,1}, \mathbf{l}_{k,2}, \mathbf{l}_{k,3}]^\top$ the arbitrary per-pixel illumination matrix.

For the re-reparameterization to be bijective, the reflectance r_k must be non-null (a basic assumption in photographic 3D vision), and L_k must be non-singular i.e., the lighting directions must be chosen linearly independent. Then, the original reflectance and normal can be retrieved from the simulated intensities by $r_k = \|\mathsf{L}_k^{-1}\mathbf{v}(\mathbf{n}_k, r_k)\|$ and $\mathbf{n}_k = \frac{\mathsf{L}_k^{-1}\mathbf{v}(\mathbf{n}_k, r_k)}{\|\mathsf{L}_k^{-1}\mathbf{v}(\mathbf{n}_k, r_k)\|}$. Considering $n > 3$ illumination vectors and resorting to the pseudo-inverse operator might induce more robustness but at the price of losing bijectivity and thus not entirely relying on the PS inputs. We leave this as a possible future work, which might be particularly interesting when the PS inputs are uncertain, or when considering more evolved PBR models involving additional reflectance clues such as roughness, anisotropy or specularity.

In practice, the choice of each arbitrary triplet of light directions $\mathbf{l}_{k,1}, \mathbf{l}_{k,2}, \mathbf{l}_{k,3}$ can be made to minimize the uncertainty on the normal estimate. To this end, the illumination triplet proposed in [4] can be considered. Therein, the authors show that the optimal configuration for three images is vectors that are equally spaced in tilt by 120 degrees, with a constant slant of 54.74 degrees (wrt to $\mathbf{n}_k$).

Let us remark that with the above linear model, it is possible to simulate negative radiance values, when one of the dot products between the normal and the lighting vectors is negative, which corresponds to self-shadowing. While negative radiance values are obviously non physically plausible, this is not a problem for the proposed re-parameterization, as long as it remains consistent with the NVR strategy, which we are now going to detail.

3.3. Volume rendering-based 3D reconstruction

We now turn our attention to deriving the volume rendering function $\tilde{\mathbf{v}}_k$ arising in Eq. (1). The role of this function is to simulate, from the scene geometry f and albedo ρ, an intensity vector $\tilde{\mathbf{v}}_k$ which will be compared with the vector $\mathbf{v}_k$ that is simulated from the inputs as described in the previous paragraph.

Our solution largely takes inspiration from the NeuS method [22], that was initially proposed as a solution to the single-light multi-view 3D surface reconstruction problem. Therein, the rendering function follows a volume rendering scheme which accumulates the colors along the ray corresponding to the k-th pixel. Denoting by $\mathbf{o}_k \in \mathbb{R}^3$ the camera center for this observation, and by $\mathbf{d}_k$ the corresponding viewing direction, this ray is written $\{\mathbf{x}_k(t) = \mathbf{o}_k + t\,\mathbf{d}_k \,|\, t \geq 0\}$. By extending the NeuS volume renderer to the multi-illumination scenario, each coefficient $\tilde{v}_{k,l}$ of $\tilde{\mathbf{v}}_k$ is then given, $\forall l \in \{1, 2, 3\}$, by:

$$\tilde{v}_{k,l} = \int_{t_n}^{t_f} w(t, f(\mathbf{x}_k(t)))\, c_l(\mathbf{x}_k(t))\, \mathrm{d}t, \tag{3}$$

where t_n, t_f stand for the range bounds over which the colors are accumulated. The weight function w is constructed from the SDF f in order to ensure that it is both occlusion-aware and locally maximal on the zero level set, see [22] for details. As for the functions $c_l : \mathbb{R}^3 \to \mathbb{R}$, they represent the scene's apparent color. In the original NeuS framework, this color depends not only on the 3D locations, but also on the viewing direction $\mathbf{d}_k$, and it is directly optimized along with the SDF f. Our case, where the albedo is optimized in lieu of the apparent color, and the illumination varies with the data index k and the illumination index l, is however slightly different.

As a major difference with this prototypical NVR-based 3D reconstruction method, we optimize the SDF f and the surface *albedo* i.e., the scene's intrinsic color ρ rather than its apparent color c_l. The dependency upon the viewing direction must thus be removed, in order to ensure consistency with the Lambertian model used for simulating the inputs. More importantly, contrarily to NeuS where the illumination is fixed, each input data $v_{k,l} := r_k \mathbf{n}_k^\top \mathbf{l}_{k,l}$ is simulated under a different, arbitrary illumination $\mathbf{l}_{k,l}$. For the NVR to produce simulations $\tilde{v}_{k,l}$ matching this input set of intensities, it is necessary to explicitly write the dependency of the apparent color c_l upon the scene's geometry f, reflectance ρ and illumination $\mathbf{l}_{k,l}$. Our volume renderer is then still given by Eq. (3), but the color of each 3D point must be replaced by:

$$c_l(\mathbf{x}_k(t)) = \rho(\mathbf{x}_k(t)) \nabla f(\mathbf{x}_k(t))^\top \mathbf{l}_{k,l}, \tag{4}$$

where the illumination vectors $\mathbf{l}_{k,l}$ are the same as those in Eq. (2).

Let us remark that the scalar product above corresponds, up to a normalization by $\|\nabla f(\mathbf{x}_k(t))\|$, to the shading. Yet, we do not need to apply this normalization, because the regularization term $\mathcal{L}_{\text{reg}}(f)$ in (1) will take care of ensuring the unit length of ∇f. Indeed, as in the original NeuS framework, the SDF is regularized using an eikonal term:

$$\mathcal{L}_{\text{reg}}(f) = \frac{\sum_{k=1}^{m} \int_{t_n}^{t_f} (\|\nabla f(\mathbf{x}_k(t))\|^2 - 1)^2 \, \mathrm{d}t}{m\,(t_f - t_n)}. \tag{5}$$

Similarly to the original NeuS, an additional regularization based on object masks can also be utilized for supervision, if such masks are provided.

Plugging (4) into (3) yields the definition of our volume renderer accounting for the varying, arbitrary illumination vectors $\mathbf{l}_{k,l}$. Next, plugging (2), (3) and (5) into (1), we obtain our objective function, which ensures the consistency between the simulations obtained from the input, and those obtained by volume rendering. It should be emphasized that, besides the eikonal regularization – which is standard and only serves to ensure the unit-length constraint of the normal, our strategy leads to a single objective optimization formulation for NVR-based 3D surface reconstruction from reflectance and normal data.

The discretization of the variational problem (1) is then achieved exactly as in the original NeuS work [22]. It is based on representing f and ρ by MLPs and hierarchically sampling points along the rays.

4. Application to MVPS

We present a standalone MVPS pipeline that is built on top of the proposed reflectance and normal-based 3D reconstruction method. Our MVPS pipeline includes the following steps:

1. Compute the reflectance and normals maps for each viewpoint through PS;
2. Select a batch of the most reliable inputs $\{r_k\}$ and $\{\mathbf{n}_k\}$;
3. Scale the reflectance values $\{r_k\}$ across the entire image collection;
4. Simulate the radiance values following Eq. (2), using a pixel-wise optimal lighting triplet L_k;
5. Optimize the loss in Eq. (1) over the SDF f and albedo ρ;
6. Reconstruct the surface from the SDF.

Step 1: PS-based reflectance and normal estimation Any PS method is suitable for obtaining the inputs for each viewpoint. However, not all PS methods actually provide reflectance clues, and not all of them can simultaneously handle non-Lambertian surfaces and unknown, complex illumination. CNN-PS [7], for instance, provides only normals, and for calibrated illumination. For these reasons, we base our MVPS pipeline on the recent transformers-based method SDM-UniPS [8], which exhibits remarkable performance in recovering intricate surface normal maps even when images are captured under unknown, spatially-varying lighting conditions in uncontrolled environments. As advised by the author of [8], when the number of images is too large for the method to be applied, one can simply take the median of the results over sufficiently many N_{trials} random trials, each trial involving the random selection of a few number of images.

Step 2: Uncertainty evaluation To prevent poorly estimated normals from corrupting 3D reconstruction, we discard the less reliable ones. To this end, we use as uncertainty measure the average absolute angular deviation of the normals computed over the N_{trials} random trials in Step 1. Pixels associated with an uncertainty measure higher than a threshold ($\tau = 15°$ in our experiments) are excluded from the optimization. Advanced uncertainty metrics, as proposed by Kaya et al. [9], could further refine this process.

Step 3: Reflectance maps scaling The individual reflectance maps computed by PS need to be appropriately scaled. This is because in an uncalibrated setting, the reflectance estimate is relative to both the camera's response, and the incident lighting intensity. Consequently, each reflectance map is estimated only up to a scale factor. To estimate this scale factor, the complete pipeline is first run without using the reflectance maps. This provides pairs of homologous points that are subsequently used to scale the reflectance maps. Concretely, given a pair of neighboring viewpoints, the ratios of corresponding reflectance values between the two viewpoints are stored, and their median is used to adjust each reflectance map's scale factor. This operation is repeated across the entire viewpoint collection. Note that, if the camera's response and the illumination were known i.e., a calibrated PS method was used in Step 1, then the reflectance would be determined without scale ambiguity and this step could be skipped.

Step 4: Radiance simulation To simulate the radiance values, we choose as lighting triplet the one which is optimal, relative to the normal $\mathbf{n}_k$ [4]. The actual formula is provided in the supplementary material.

Step 5: Optimization The actual optimization of the loss function is carried out using a straightforward adaptation of the NeuS architecture [22], where viewing direction was removed from the network's input to turn radiance into albedo. In all our experiments, we let the optimization run for a total of 300k iterations, with a batch size of 512 pixels. To ensure that the networks have a better understanding of our MVPS data, we decided to train each iteration not only on a random view, but also on all rendered images of this view under varying illumination. The backward operation is then applied only after the loss is computed on all pixels for all the illumination conditions. In terms of computation time, our approach is comparable with the original NeuS framework, requiring in our tests from 8 to 16 hours on a standard GPU for the 3D reconstruction of each dataset from DiLiGenT-MV [12].

Step 6: Surface reconstruction Once the SDF is estimated, we extract its zero level set using the marching cube algorithm [15].

5. Experimental results

5.1. Experimental setup

Evaluation datasets We used the DiLiGenT-MV benchmark dataset [12] to perform all our experiments, statistical evaluations, and ablations. It includes five real-world objects with complex reflectance properties and surface profiles, making it an ideal choice for the proposed method evaluation. Each object is imaged from 20 calibrated viewpoints using the classical turntable MVPS acquisition setup [6]. For each view, 96 images are acquired under different illuminations. Given the large volume of images, which is impractical for transformers-based methods, our implementation of Step 1 (PS) employs SDM-UniPS [8] with only 10 input images. To this end, we computed each r_k and $\mathbf{n}_k$ as the medians of the computed reflectances and normals over $N_{\text{trials}} = 100$ random trials, each trial involving the random selection of 10 images from the 96 available in the DiLiGenT-MV dataset.

Evaluation scores We performed our quantitative evaluations using F-score and Chamfer distance (CD), to measure the accuracy of the reconstructed vertices. We also measured the mean angular error (MAE) of the imaged meshes, to evaluate the accuracy of the reconstructed normals wrt the ground truth normals provided in DiLiGenT-MV. We report both the results averaged over all mesh vertices, and those on vertices clustered in two particularly interesting classes, namely high curvature and low visibility areas, as illustrated in Fig. 3. To identify the high curvature areas, we used the library VCGLib [1] and the 3D mesh processing software system Meshlab [3], taking the absolute value of the curvature to merge the convex and concave zones and retaining the vertices whose curvature is higher than 1.6. To segment the low visibility areas, we summed the boolean visibility of each vertex in each view. Low visibility then corresponds to vertices visible in less than 5 viewpoints, among the 20 ones of DiLiGenT-MV.

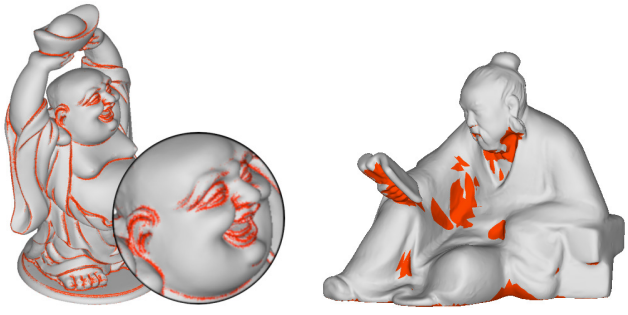


Figure 3. High curvature (left) and low visibility (right) areas, on the Buddha and Reading datasets.

5.2. Baseline comparisons

We first provide in Fig. 4 a qualitative comparison of our results on four objects, and compare them with the three most recent methods from the literature, namely PS-NERF [26], Kaya23 [11] and MVPSNet [27]. In comparison with these state-of-the-art deep learning-based methods, the recovered geometry is overall more satisfactory.

This is confirmed quantitatively when evaluating Chamfer distances and MAE, provided in Tables 1 and 2. Therein, beside the aforementioned methods we also report the results from the Kaya22 method [9] and those from the non deep learning-based ones Park16 [20] and Li19 [12] (which is not fully automatic). From the tables, it can be seen that our method outperforms other fully automated standalone ones, and is competitive with the semi-automated one. On average, our method reports a Chamfer distance which is 17.4% better than the second best score, obtained by MVPSNet [27]. Regarding MAE, our score is similar to Kaya23 [11] with a small average difference of 0.2 degree. The superiority of our approach can also be observed by considering the F-scores, which are reported in Fig. 5.

	Chamfer distance ↓					
Methods	Bear	Budd.	Cow	Pot2	Read.	Aver.
Park16	0.92	0.39	0.34	0.94	0.53	0.62
Li19 †	0.22	0.28	0.11	0.23	0.27	0.22
Kaya22	0.39	0.4	0.3	0.4	0.35	0.37
PS-NeRF	0.32	0.28	**0.24**	**0.24**	0.33	0.28
Kaya23	0.33	**0.21**	**0.22**	0.37	0.28	0.28
MVPSNet	**0.28**	0.3	0.25	0.27	**0.25**	**0.27**
Ours	**0.22**	**0.22**	0.25	**0.16**	**0.27**	**0.23**

Table 1. Chamfer distance (lower is better) averaged overall all vertices. **Best results**. **Second best**. Since † requires manual efforts, it is not ranked.

	Normal MAE ↓					
Methods	Bear	Budd.	Cow	Pot2	Read.	Aver.
Park16	9.64	12.6	8.23	11.1	9.01	10.1
Li19 †	3.85	11.0	2.82	5.88	6.30	5.97
Kaya22	4.89	12.5	4.44	8.68	6.52	7.41
PS-NeRF	5.48	11.7	5.46	7.65	9.13	7.88
Kaya23	**3.24**	**8.12**	**3.04**	5.63	**5.66**	**5.14**
MVPSNet	5.26	14.1	6.28	6.69	8.58	8.18
SDM-UniPS*	4.79	9.60	5.46	**5.56**	10.1	7.12
Ours	**2.70**	**8.17**	**3.61**	**4.11**	**6.18**	**4.95**

Table 2. Normal MAE (lower is better) averaged over all views. For reference, the mono-view PS results from SDM-UniPS [8] (*) are also provided, although it does not provide a full 3D reconstruction and thus its Chamfer distance cannot be evaluated.

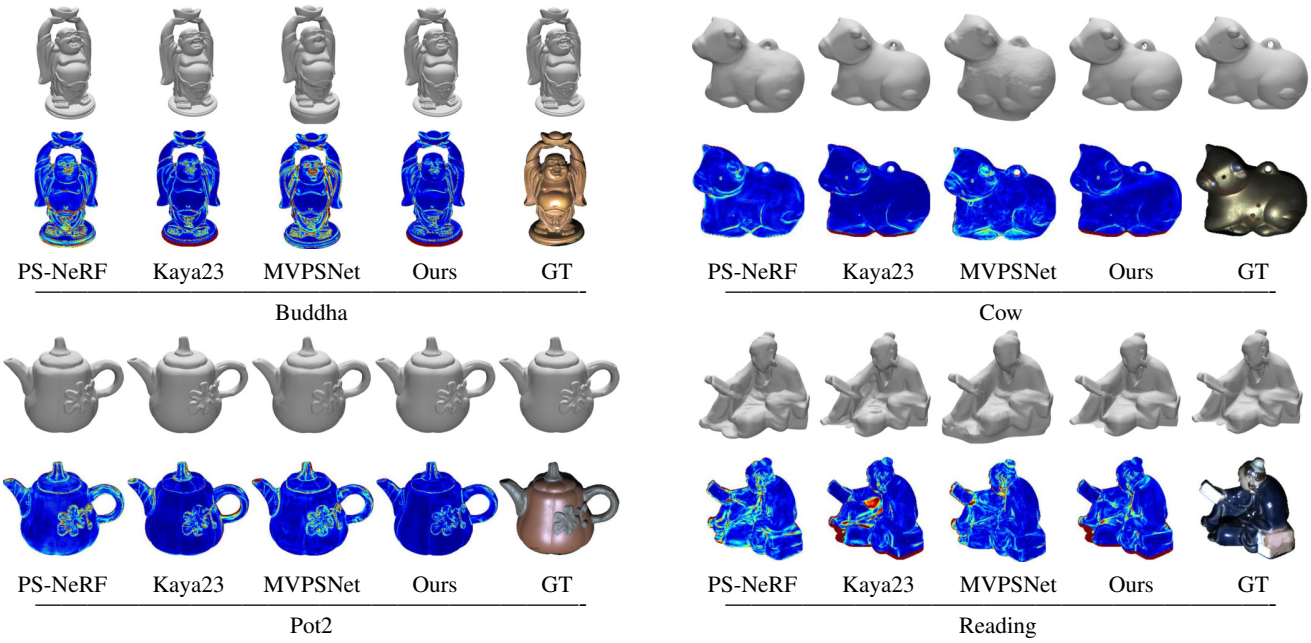
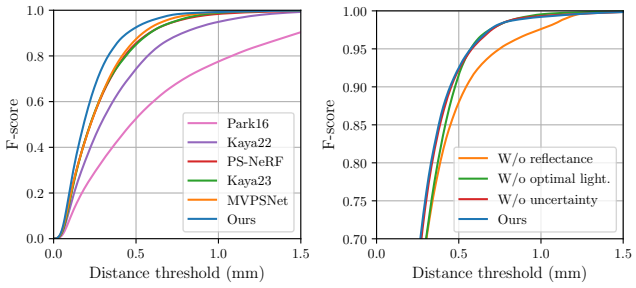


Figure 4. Reconstructed 3D mesh and corresponding angular error of four objects from the DiLiGenT-MV benchmark.



(a) (b)

Figure 5. F-score (higher is better) as a function of the distance error threshold, in comparison with other state-of-the-art methods (a), and disabling individual components of our method (b).

	All		High curv.		Low vis.	
% Vertices	100%		8.27%		8.70%	
Scores	CD	MAE	CD	MAE	CD	MAE
Park16	0.62	10.1	0.88	29.0	0.68	29.6
Li19 †	0.22	5.97	0.51	26.2	0.67	33.3
Kaya22	0.37	7.41	0.45	28.0	0.54	31.7
PS-NeRF	0.28	7.88	0.38	25.8	0.5	24.0
Kaya23	0.28	**5.14**	**0.29**	**23.6**	**0.41**	**20.7**
MVPSNet	**0.27**	8.18	0.53	23.9	0.49	28.9
Ours	**0.23**	**4.95**	**0.24**	**23.1**	**0.26**	**17.8**

Table 3. Chamfer distance and normal MAE (lower is better) on high curvature and low visibility areas.

5.3. High curvature and low visibility areas

To highlight the level of details in the 3D reconstructions, Figs. 1 and 6 provide other qualitative comparisons focusing on one small part of each object. Ours is the only method achieving a high fidelity reconstruction on the ear, the knot and the navel of Buddha, and on the spout of Pot2. To quantify this gain, we also report in Table 3 the average CD and MAE over all datasets, yet taking into account only the high curvature and low visibility areas. It is worth noticing that the CD error of PS-NeRF and MVPSNet on high curvature areas increases by 36% and 96%, respectively, in comparison with that averaged over the entire set of vertices. Ours, on the contrary, increases by 4% only. Similarly, on low visibility areas their error increases by 78% and 81%, and Kaya23 by 46%, while ours increases only by 13%.

5.4. Ablation study

Lastly, we conducted an ablation study, to quantify the impact of some parts of our pipeline. More precisely, we quantify in Fig. 5b and Table 4 the impact of providing PS-estimated reflectance maps, in comparison with providing only normals ("W/o reflectance"). We also evaluate that of the pixel-wise optimal lighting triplet, in comparison with using the same arbitrary one for all pixels in one view ("W/o optimal lighting"). Lastly, we evaluate the impact of discarding the less reliable inputs, in comparison with using all of them ("W/o uncertainty"). The feature that influences most the accuracy of the 3D reconstruction is the use of reflectance. The other two features also positively impact the reconstruction, but to a lesser extent.

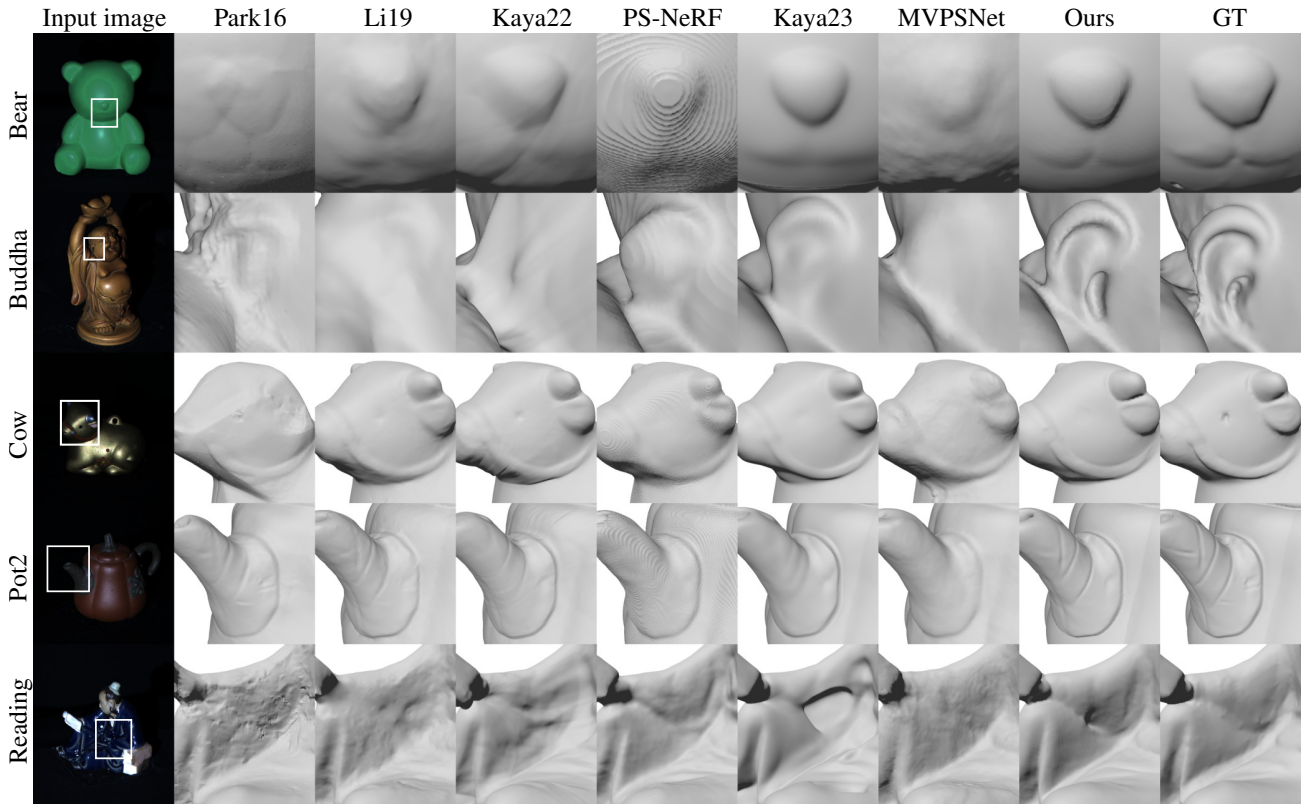


Figure 6. Qualitative comparison between our results and state-of-the-art ones, on parts of the meshes representing fine details.

	Chamfer distance ↓					
Methods	Bear	Budd.	Cow	Pot2	Read.	Aver.
W/o reflect.	0.23	0.22	0.39	0.16	0.31	0.26
W/o opt. l.	0.32	**0.22**	**0.20**	0.19	**0.27**	0.24
W/o uncert.	0.22	0.22	0.27	0.16	0.27	0.23
Ours	**0.22**	0.22	0.25	**0.16**	0.27	**0.23**

Table 4. Chamfer distance (lower is better) averaged overall all vertices, while disabling individual features of the pipeline (reflectance estimation, optimal lighting, and uncertainty evaluation).

5.5. Limitations

Our approach heavily relies on the quality of the PS normal maps. In our experiments, we used SDM-UniPS [8], which generally yields high quality results. Yet, it occasionally yields corrupted normals, leading to inconsistencies across viewpoints that may result in errors in the reconstruction (cf. supplementary material). This could be handled in the future by replacing the PS method by a more robust one. A second limitation, similar to PS-NeRF, is the computation time, which falls within the range of 8 to 16 hours for one object in DiLiGenT-MV. Fortunately, NeuS2 [23], a significantly faster version of NeuS, will allow us to reduce the computation time to around ten minutes.

6. Conclusion

We have introduced a neural volumetric rendering method for 3D surface reconstruction based on reflectance and normal maps, and applied it to multi-view photometric stereo. The proposed method relies on a joint re-parameterization of reflectance and normal as a vector of radiances rendered under simulated, varying illumination. It involves a single objective optimization, and it is highly flexible since any existing or future PS method can be used for constructing the input reflectance and normal maps. Coupled with a state-of-the-art uncalibrated PS method, our method reaches unprecedented results on the public dataset DiLiGenT-MV in terms of F-score, Chamfer distance and mean angular error metrics. Notably, it provides exceptionally high quality results in areas with high curvature or low visibility. Its main limitation for now is its computational cost, which we plan to reduce by adapting recent developments within the NeuS2 framework [23]. Using reflectance uncertainty in addition to that of normal maps offers room for improvement.

Acknowledgements. This work was supported by the Danish project PHYLORAMA, the ALICIA-Vision project, the IMG project (ANR-20-CE38-0007), the OR-X and associated funding by the University of Zurich and University Hospital Balgrist.

References

[1] VCGLib. https://github.com/cnr-isti-vclab/vcglib. 6

[2] Meghna Asthana, William Smith, and Patrik Huber. Neural apparent BRDF fields for multiview photometric stereo. In *Proceedings of the 19th ACM SIGGRAPH European Conference on Visual Media Production*, pages 1–10, 2022. 2

[3] Paolo Cignoni, Marco Callieri, Massimiliano Corsini, Matteo Dellepiane, Fabio Ganovelli, Guido Ranzuglia, et al. Meshlab: an open-source mesh processing tool. In *Proceedings of the Eurographics Italian Chapter Conference*, pages 129–136, 2008. 6

[4] Ondrej Drbohlav and Mike Chantler. On optimal light configurations in photometric stereo. In *Proceedings of the 10th IEEE International Conference on Computer Vision*, pages 1707–1712, 2005. 4, 5

[5] Yasutaka Furukawa, Carlos Hernández, et al. Multi-view stereo: A tutorial. *Foundations and Trends® in Computer Graphics and Vision*, 9(1-2):1–148, 2015. 1

[6] Carlos Hernández, George Vogiatzis, and Roberto Cipolla. Multiview Photometric Stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):548–554, 2008. 2, 6

[7] Satoshi Ikehata. CNN-PS: CNN-based photometric stereo for general non-convex surfaces. In *Proceedings of the European Conference on Computer Vision*, pages 3–18, 2018. 2, 5

[8] Satoshi Ikehata. Scalable, Detailed and Mask-Free Universal Photometric Stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13198–13207, 2023. 1, 2, 5, 6, 8

[9] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Uncertainty-aware deep multi-view photometric stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12601–12611, 2022. 2, 5, 6

[10] Berk Kaya, Suryansh Kumar, Francesco Sarno, Vittorio Ferrari, and Luc Van Gool. Neural radiance fields approach to deep multi-view photometric stereo. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1965–1977, 2022. 2

[11] Berk Kaya, Suryansh Kumar, Carlos Oliveira, Vittorio Ferrari, and Luc Van Gool. Multi-View Photometric Stereo Revisited. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3126–3135, 2023. 1, 2, 6

[12] Min Li, Zhenglong Zhou, Zhe Wu, Boxin Shi, Changyu Diao, and Ping Tan. Multi-view photometric stereo: A robust solution and benchmark dataset for spatially varying isotropic materials. *IEEE Transactions on Image Processing*, 29:4159–4173, 2020. 1, 2, 5, 6

[13] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-Fidelity Neural Surface Reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8456–8465, 2023. 1

[14] Fotios Logothetis, Roberto Mecca, and Roberto Cipolla. A differential volumetric approach to multi-view photometric stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1052–1061, 2019. 2

[15] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field*, pages 347–353. 1998. 5

[16] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[17] Diego Nehab, Szymon Rusinkiewicz, James Davis, and Ravi Ramamoorthi. Efficiently combining positions and normals for precise 3D geometry. *ACM Tansactions on Graphics*, 24 (3):536–543, 2005. 2

[18] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 2

[19] Jaesik Park, Sudipta N Sinha, Yasuyuki Matsushita, Yu-Wing Tai, and In So Kweon. Multiview photometric stereo using planar mesh parameterization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1161–1168, 2013. 2

[20] Jaesik Park, Sudipta N Sinha, Yasuyuki Matsushita, Yu-Wing Tai, and In So Kweon. Robust multiview photometric stereo using planar mesh parameterization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(8): 1591–1604, 2016. 2, 6

[21] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14194–14203, 2021. 2

[22] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. In *Proceedings of the Conference on Neural Information Processing Systems*, 2021. 1, 3, 4, 5

[23] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3295–3306, 2023. 8

[24] Robert J Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980. 1

[25] Qingshan Xu, Weihang Kong, Wenbing Tao, and Marc Pollefeys. Multi-scale geometric consistency guided and planar prior assisted multi-view stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4945–4963, 2022. 1

[26] Wenqi Yang, Guanying Chen, Chaofeng Chen, Zhenfang Chen, and Kwan-Yee K Wong. PS-NeRF: Neural Inverse Rendering for Multi-view Photometric Stereo. In *Proceedings of the European Conference on Computer Vision*, pages 266–284, 2022. 1, 2, 6

[27] Dongxu Zhao, Daniel Lichy, Pierre-Nicolas Perrin, Jan-Michael Frahm, and Soumyadip Sengupta. MVPSNet: Fast Generalizable Multi-view Photometric Stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12525–12536, 2023. 1, 2, 6