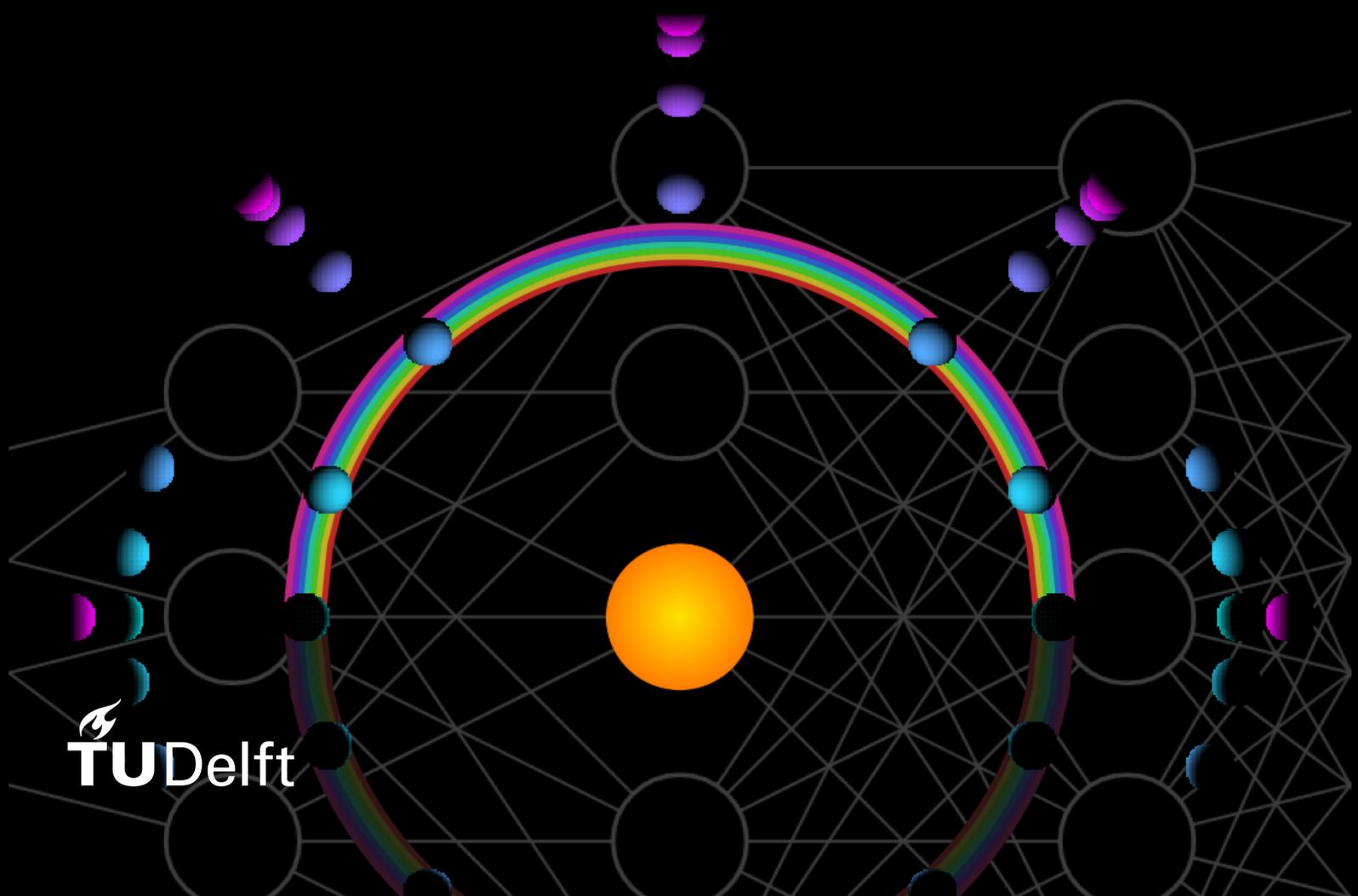


Master's Thesis

Neural Networks for Exoplanet Cartography

Klaas Meinke



Neural Networks for Exoplanet Cartography

by

Klaas Meinke

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Tuesday June 22, 2021 at 3:00 PM.

Student number:	4536053	
Thesis committee:	Dr. D.M. Stam	TU Delft, Supervisor
	Dr. P.M. Visser	TU Delft, Supervisor
	Prof. P.N.A.M. Visser	TU Delft
	Dr. J. Sun	TU Delft

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.



Abstract

By the late 2020s or early 2030s, the next generation of telescopes will be able to directly observe the reflected starlight of Earth-like exoplanets. Because of the huge distance to other stars, such exoplanets will appear as single unresolved pixels. A single pixel can, however, provide information about what the planet looks like because its brightness varies in time as it rotates about its axis and orbits its star. Several researchers have shown that these changes in brightness can, indeed, be used to retrieve a map of the planet. Their methods use the Lambertian model of diffuse reflection to retrieve albedo maps of the planet's surface. We aim to develop new algorithms that can retrieve maps of non-Lambertian planets with a Rayleigh scattering atmosphere, clouds with water droplets that cause rainbows and oceans that exhibit a glint feature. We also aim to evaluate the validity of the Lambertian assumption for such non-Lambertian planets. We numerically compute reflected light curves with the previously mentioned non-Lambertian effects, including the polarization of the reflected light. Instead of retrieving albedo maps, we classify facets by their surface type and cloud coverage, using convolutional neural networks. We show that a convolutional neural network can classify facets on a non-Lambertian planet with an accuracy of 87% for an ideal geometry and no noise, when the rotation axis is known. Using another neural network architecture, we show that the rotation axis can be constrained with a mean squared error as low as 0.006 for our training data and similar results are seen for a model Earth. Including polarization in the retrieval improves the rotation axis retrieval's mean squared error (MSE) by roughly 15% and the classification accuracies of ocean facets and cloudy facets by 2% and 1% respectively. We show that a retrieval algorithm that relies on the Lambertian assumption causes concentric artefacts about the poles when applied to light curves of a non-Lambertian planet for all inclinations besides face-on. The MSE of the rotation axis retrievals increases by roughly one order of magnitude for these inclination when making the Lambertian assumption.

Delft, June 2021

Contents

1	Introduction	1
2	Creating Planets	4
2.1	Planet Facets	4
2.1.1	HEALPix	5
2.1.2	Fibonacci Sphere	6
2.1.3	Comparison of Facet Schemes	7
2.2	Surface Maps	7
2.2.1	Elevation Profile	7
2.2.2	Surface Types	8
2.3	Cloud Maps	10
2.4	Inclinations	11
2.5	Rotation Axes	12
2.6	Combinations	12
3	Computing Reflected Fluxes	14
3.1	Reflection Models	15
3.1.1	Directional Reflection	15
3.1.2	Lambertian Reflection	17
3.1.3	Phase Curve Comparison	17
3.1.4	Resolved Flux Comparison	19
3.2	Orbit	19
3.3	Rotation	22
3.4	Creating a Large Data Set	23
3.5	Noise	23
3.5.1	Shot Noise	23
3.5.2	Shot Noise of Planned Exoplanet Missions	25
3.6	Normalization	27
4	Rotation Axis Retrieval	28
4.1	Architecture	28
4.1.1	Periodic Convolutions	28
4.1.2	Inclination Input	29
4.1.3	Final Architecture	30
4.2	Degeneracy	31
4.3	Training	32
4.3.1	Training Parameters	32
4.3.2	PReLU Slope Coefficients	32
4.4	Numerical Artefacts	33
4.5	Retrieval Accuracy	33
4.5.1	With Degeneracy	33
4.5.2	Without Degeneracy	35
4.6	Verification	36
5	Albedo Map Retrieval	37
5.1	Absolute Light Curves	37
5.2	Retrieval Accuracies for Different Inclinations and Rotation Axes	39
5.3	Relative (Normalized) Light Curves	40
5.4	Effects of Noise	40
5.5	Errors due to Lambertian Assumption	42

6	Surface Type Map Retrieval	44
6.1	Architecture	44
6.1.1	Spherical Convolutions	44
6.1.2	Architecture	45
6.1.3	Training	45
6.2	Retrieval Accuracy	46
6.3	Validity of Lambertian Assumption	47
6.4	Benefits of Polarization	49
7	Conclusion	51
8	Recommendations	53
	Bibliography	54

1

Introduction

Since 1995, when Mayor and Queloz [29] discovered the first exoplanet orbiting another star, more than 4000 exoplanets have been identified and catalogued.¹ Nowadays, statistics show that nearly all stars have exoplanets orbiting around them (Dressing and Charbonneau [5], Tuomi et al. [40]). Some of these exoplanets lie in the "habitable zone", close enough to their star that, assuming the presence of an Earth-like atmosphere, water on the surface does not freeze and far away enough that it does not boil off, instead forming oceans. These Earth-like landscapes of continents and water oceans are a promising environment for extraterrestrial life to evolve.

With the aim of finding signs of extraterrestrial life, NASA and ESO are currently designing the next generation of telescopes that will be able to detect reflected starlight from these Earth-like exoplanets. This is particularly difficult due to the very faint exoplanet signal and the adjacent starlight that is at least 10 million times brighter (Hunziker et al. [18]). The approach taken by the Starshade Rendezvous Probe mission (Sara Seager [33]) and HabEx (JPL [19]) is to fly a space telescope in formation with a flower-shaped starshade that blocks the bright starlight, an idea originally proposed by Cash [4]. Another approach is the European Extremely Large Telescope (E-ELT),² which is large enough to observe Earth-like exoplanets about some stars from the ground using a coronagraph, which blocks the bright starlight from inside the telescope, in combination with an extreme adaptive optics system. Both approaches are expected to take the first images of Earth-like exoplanets in the late 2020s or early 2030s. Due to the huge distance, however, the exoplanets will appear as single pixels, much like Earth in the Pale Blue Dot picture taken by Voyager 1 as it left the solar system (Figure 1.1).



Figure 1.1: The Pale Blue Dot picture of Earth, taken by Voyager 1 as it left the solar system. Like Earth in this picture, a directly observed exoplanet will appear as a single pixel. In this thesis, we use the pixel's varying brightness to infer the planet's map.³

¹<https://exoplanetarchive.ipac.caltech.edu/>

²<https://www.eso.org/public/teles-instr/elt/>

³Image Source: <https://solarsystem.nasa.gov/resources/536/voyager-1s-pale-blue-dot/>

Also due to the huge distances, the intensity of the received light will be very low. Even with the largest near-future telescopes, the number of photons received from the planet will need to be counted for minutes, hours or days to obtain a clear measurement. In Figure 1.2, we show the number of photons that would be received by the Starshade Rendezvous Probe (with the Roman Space Telescope) and HabEx, which have aperture diameters of 2.4 and 4 meters respectively, for a completely white, Earth-sized planet around a Sun-type star at full phase. We assume that the flux is integrated over 3 hours and wavelengths within a range of 50 nm are combined. For these low levels of photons, the particle nature of light gives rise to a noise called shot noise, so that even a perfect instrument with no background noise cannot make perfect observations of the planet's brightness.

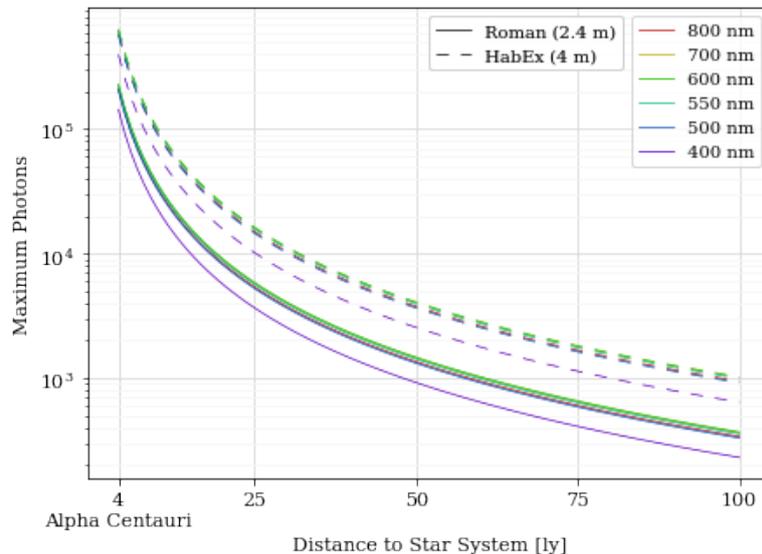


Figure 1.2: The maximum number of photons (white Lambertian planet at full phase) for an Earth-sized planet around a Sun-type star. The integration time is 3 hours and the wavelength bands are 50 nm wide such that each wavelength can be resolved. HabEx is a proposed mission with a starshade and dedicated telescope for the 2030s (JPL [19]) and the Nancy Grace Roman Space Telescope, to be launched in the mid-2020s,⁴ has hardware to fly in formation with a starshade that could be launched in the late 2020s (Sara Seager [33]). Lines for some wavelengths are covered by other wavelengths.

If the planet's surface is not homogeneous due to continents, oceans and/or clouds, its brightness will vary as it rotates about its axis and different surfaces are illuminated by the starlight. These daily variations are caused by longitudinal variations on the planet and can be seen by the high-frequency oscillations in Figure 1.3. At larger timescales, the planet's orbit about its star causes phase changes, much like the ones we see on the moon. Since the planet is unresolved, however, we don't see a change in the shape, as we do for the moon, but a change in the brightness of the unresolved pixel. The brightness variations due to phase changes are on an annual timescale and are caused by latitudinal and longitudinal variations on the planet, depending on the orientation of the planet's orbit and rotation axis. One final variation that can be observed is the polarization (the direction of the electric field vector) of the reflected light waves. Depending on the directions of the incoming and outgoing light rays, Rayleigh scattering in the atmosphere, scattering by clouds particles and reflection by surfaces like oceans can cause the planetary signal to become polarized such that the light waves tend to be oriented in a particular direction, which can be measured by a polarimeter.

Some authors (for example Fujii and Kawahara [8], Kawahara and Masuda [23], Fan et al. [6], Farr et al. [7] and Asensio Ramos and Pallé [2]) have started solving the problem of creating hypothetical maps of planets using numerically simulated observations of reflected starlight. They have shown that by measuring the changes in brightness of a planet over time, it is possible to constrain its rotation axis and retrieve a rough planetary surface map, using daily variations for longitudinal resolution and annual variations for latitudinal resolution. In these studies, the planets' surfaces are treated as Lambertian reflectors, which diffusely reflect starlight in all directions.

We propose a retrieval algorithm which uses the more accurate model of directionally dependent reflection, described by Rossi et al. [32]. The model includes an Earth-like atmosphere with Rayleigh scattering,

⁴<https://roman.gsfc.nasa.gov/>

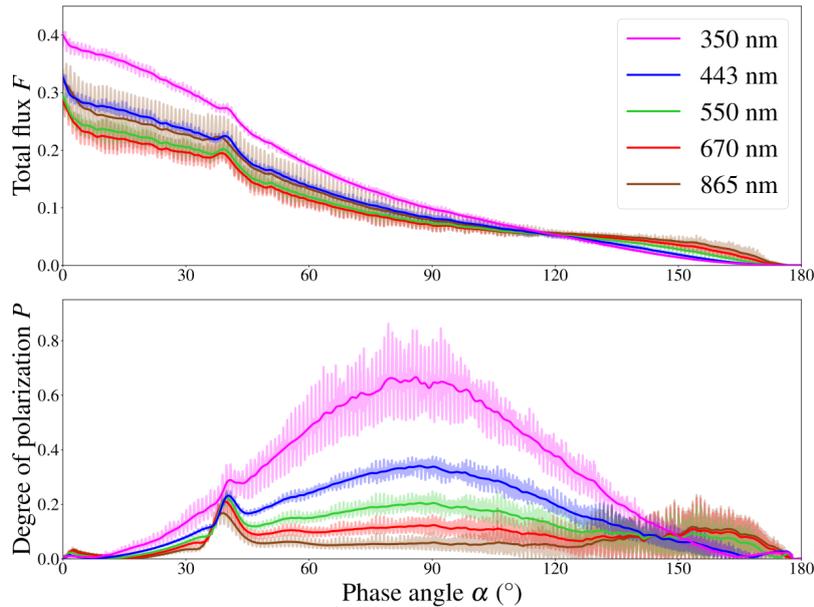


Figure 1.3: The variations in brightness, F , and degree of polarization, P , of a model Earth throughout half of an orbit. High-frequency daily fluctuations are due to the rotation of the planet about its axis and longer scale variations are due the movement from directly behind the star ($\alpha = 0^\circ$) to directly in front of the star ($\alpha = 180^\circ$). These flux curves are created by Groot et al. [12] using the same directional reflection models used in this thesis.

water clouds that create a rainbow at certain orbital locations and oceans that exhibit a bright glint feature. Ocean glint is the same phenomenon you observe when you see the reflection of the sun in the ocean at the beach and is shown by Trees and Stam [39] to significantly affect a planet's brightness. The polarization of the reflected light is also dependent on the direction of reflection, so this is also included in the model and used for the retrievals.

Using light curves created with this more accurate model requires a new approach to planet mapping in which the surface types are retrieved, rather than albedo values. We approach this problem by using neural networks, which has several advantages: Neural networks are universal approximators (Hornik [17]) and can thus be used to solve the non-linear problem of constraining a planet's rotation axis and neural networks have achieved state-of-the-art-results in classification problems (Schmidhuber [34]), which is the nature of surface type mapping. Since directional reflection is more complicated than Lambertian reflection, it seems that analytical methods would be difficult to apply to the more accurate light curves so neural networks may be the better solution.

The surface type maps we achieve in this research using neural networks are of similar detail to the albedo maps retrieved by other authors for Lambertian planets, with more than 85% of the planet being correctly classified in the ideal case. We show that including a polarimeter and measuring polarization on the next generation of telescopes would somewhat improve the retrieval of both the rotation axes and maps of observed Earth-like exoplanets. We also show that the Lambertian assumption causes large errors for inclinations besides a face-on orbit.

In Chapter 2 of this thesis, we create a set of 4 million fake Earth-like planets, with different surface and clouds maps, that can be used to train neural networks. The next step, in Chapter 3, is to compute the reflected flux of each planet for different orbital locations and rotation phases. We design, test and train a neural network to infer the planets' rotation axes based on these light curves in Chapter 4. In Chapter 5, we recreate other authors' results by retrieving albedo maps of Lambertian planets using simple neural networks. We show that when a retrieval algorithm based on the Lambertian assumption is applied to non-Lambertian planets, strong concentric artefacts about the poles can result. We present the new method of mapping, by classifying facets on the planetary surface as different surface types in Chapter 6. Finally, the conclusions and recommendations for future research can be found in Chapter 7 and Chapter 8, respectively.

2

Creating Planets

The goal of this thesis is to create retrieval algorithms that infer an exoplanet's surface map based on direct observations. The neural network approach requires a large set of training data, consisting of many planets' properties and their respective observations. Ideally this training data would be taken from actual planets and moons but this is impossible since the planets and moons in our solar system are too few and not sufficiently Earth-like. In this chapter, we thus create a set of fake planets, for which synthetic observations are simulated in Chapter 3.

Four parameters are varied between the planets:

- Cloud Map
- Surface Map
- Orbital Inclination
- Rotation Axis.

To reduce computation and memory requirements, each planet does not have four unique parameters. Instead, a set of possibilities is defined for each parameter. Then each planet is created by randomly drawing from the sets of possibilities and stored as four indices. 4 million planets are thus created nearly instantly.

The first step in defining cloud and surface maps is defining facets on the surface of the planet in Section 2.1. Next, Section 2.2 - Section 2.5 describe how each set of parameters is created, respectively. The goal of these methods is an unbiased set of possibilities, as peculiarities of the data set can be learned by the networks. To this end, planet maps that are similar to Venus (completely cloud covered) and Mars (nearly completely desert covered) are included in the training set and there is no bias in the inclination or tilt of the planets. The final set of planets is presented in Section 2.6.

2.1. Planet Facets

To simulate the reflected star light of a planet, the planet's surface should be defined by either continuous functions or a set of discrete facets. The facet approach allows for simple distinction between different surface types such as those seen on the terrestrial planets and moons in our Solar System. For this reason, it has been chosen for this thesis research. We identify several requirements for the surface facets:

- The set of facets should approximate a spheroid.
- The set of facets should be three dimensional, so that rotation about the axis and the star is possible.
- The set of facets should be approximately equal in area so that the level of detail on each part of the planet is approximately equal.
- The number of facets on the sphere should be variable so that the spatial resolution can be adapted if necessary.

We have investigated two types of facet schemes that (partially) fulfill these requirements: the HEALPix scheme developed by Gorski et al. [11] and the Fibonacci lattice by González [10].

2.1.1. HEALPix

HEALPix (Hierarchical Equal Area isoLatitude Pixelization) is a facet scheme developed for full-sky cosmological maps and it is the standard scheme used for exoplanet cartography (see for example Fujii and Kawahara [8], Kawahara and Masuda [23], Fan et al. [6], Farr et al. [7], Asensio Ramos and Pallé [2]). Besides the facets' equal areas, a major advantage of the HEALPix scheme is that nearby facets are indexed consecutively, allowing for easier data processing for "convolutions with local and global kernels, Fourier analysis with spherical harmonics and power spectrum estimation, wavelet decomposition," etc. (Gorski et al. [11]).

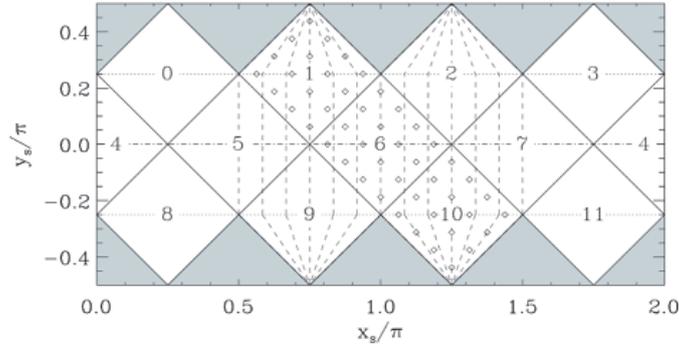


Figure 2.1: The 12 base resolution facets of HEALPix, one of the two considered facet schemes. All facets have an equal area on the surface of a sphere and can be subdivided for higher resolutions as done in Figure 2.2 for a total of 768 facets. (Figure 5 from Gorski et al. [11])

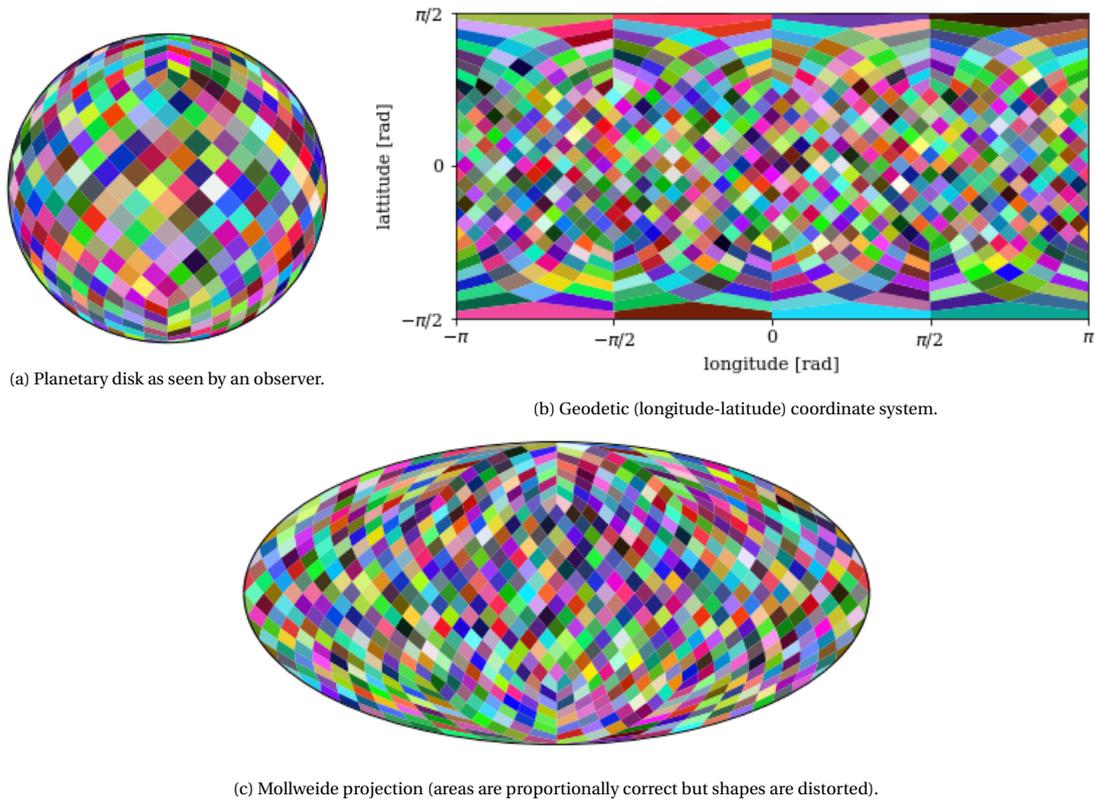


Figure 2.2: A higher resolution HEALPix scheme, created by subdividing the base facets in Figure 2.1 into four equal rhombuses three times for a total of $12 \cdot 4^3 = 768$ facets.

The HEALPix scheme is based on 12 base facets, as shown in Figure 2.1. In order to achieve higher resolutions, the base facets are each subdivided into four quadrilaterals, allowing for $12 \cdot 4^N$ total facets for any non-negative integer N . A HEALPix scheme divided three times ($N = 3$), thus resulting in 768 facets, is shown for different projections in Figure 2.2.

2.1.2. Fibonacci Sphere

The Fibonacci spiral is a method of creating any positive integer N of evenly spaced points on the surface of a sphere, based on the golden ratio. One axis through the sphere is chosen (in our case the y -axis) and for a sphere with radius R the coordinates of the sampled points are evenly spaced in the range $-R \leq y \leq R$. The other two Cartesian coordinates (in our case x and z) are found by rotating about the chosen axis by the golden ratio ($\approx 137.5^\circ$), as shown in Figure 2.3. Due to the nature of the golden ratio, periodicities or near-periodicities are avoided and so there is no clumping of the points (González [10]).

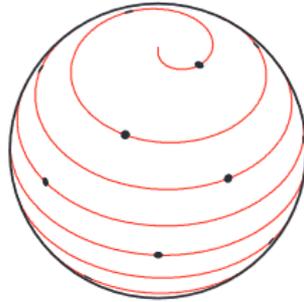


Figure 2.3: The Fibonacci Spiral with 21 equally spaced points on the surface of a sphere, created by equally distributing along one axis and rotating by the golden ratio about said axis. (Figure 2 from González [10].)

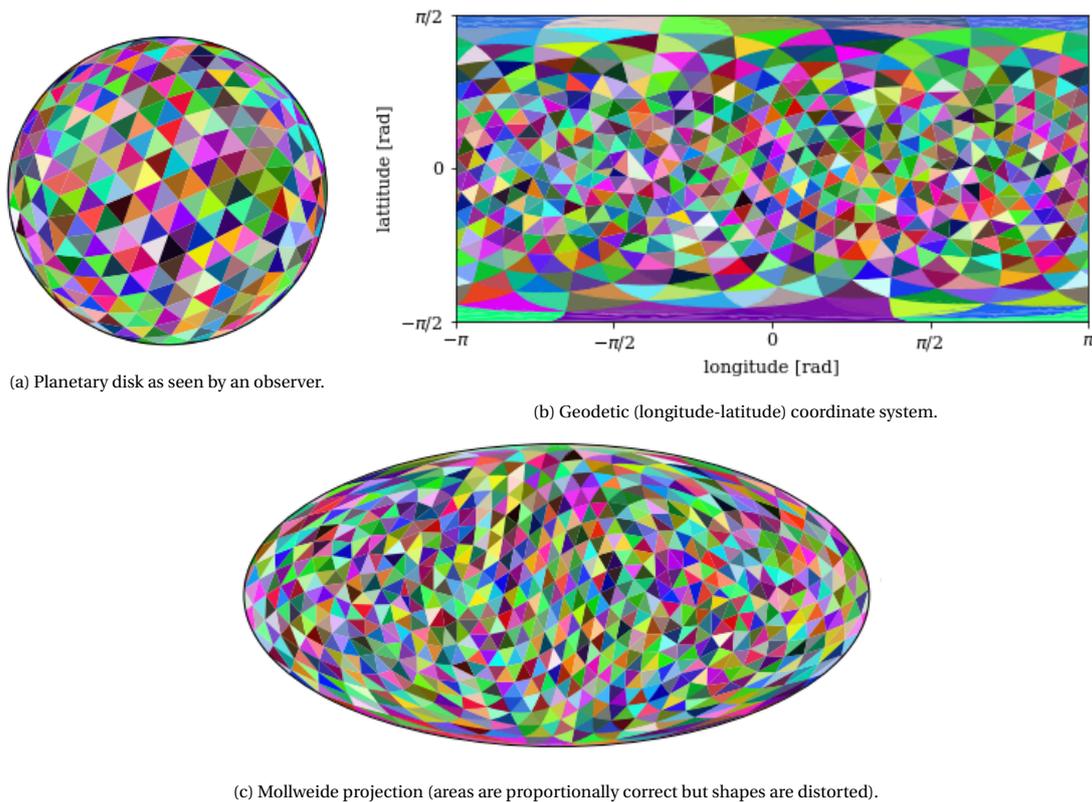


Figure 2.4: A Fibonacci sphere with 768 facets, created by passing 383 points of a Fibonacci spiral into the scipy ConvexHull function.

To convert the sampled points into a set of facets that approximate a sphere, the convex hull problem needs to be solved, which is the problem of identifying the lines (in 2-D) or facets (in 3-D) which encapsulate a set of points while only using those points as vertices. To do this, the ConvexHull function from the SciPy Python package, which is an implementation of the Quickhull algorithm, is used.⁵ This splits the sphere into

⁵<https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.ConvexHull.html>

$2N+2$ triangular facets when there are N sampled points.

The resulting facet scheme for 768 facets is shown in Figure 2.4 and shows that the facets are far more irregular than when using the HEALPix scheme and patterns are difficult to recognize. The facets appear to be roughly equal in size, which is confirmed by plotting the facets' areas in Figure 2.5, which shows that their standard deviation is only 6% of their mean.

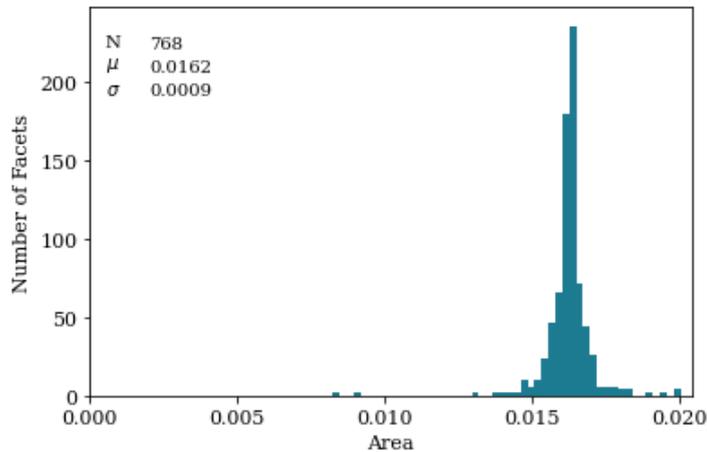


Figure 2.5: Areas of the facets on the Fibonacci sphere shown in Figure 2.4. N , μ and σ indicate the total number, mean and standard deviation, respectively. The areas are verified by multiplying N and μ and a deviation of 0.8% from the surface area of a perfect sphere is found (radius = 1). The two left-most outliers are faces with one vertex at $[0,1,0]$.

2.1.3. Comparison of Facet Schemes

Both of the first two requirements for the facet scheme, which are approximation of a spheroid and three-dimensionality, are fully met by both HEALPix and the Fibonacci method. The third and fourth requirements state that the areas of the facets should be roughly equal and that the number of facets should be easy to vary, respectively. HEALPix has equal facet areas, but can only be used for $12 \cdot 4^N$ (12, 48, 192, 768, 3072 ...) number of facets. The Fibonacci sphere has only approximately equal facet areas but can be used for any even number of facets. We decide to use the Fibonacci sphere since the shortcomings of HEALPix seem to outweigh those of the Fibonacci sphere, which has nearly equal facet areas. The ability to vary the number of facets when using the Fibonacci method is helpful for creating fake planets and trying different network architectures to retrieve planet maps. Through trial and error, we find that using 1000 facets strikes a balance between sufficient detail in the spatial resolution and efficient training of neural networks.

2.2. Surface Maps

After splitting the planetary surface into facets, each facet can be assigned a surface type such as the ones we see on Earth (ocean, vegetation, desert, etc.), to create maps of Earth-like planets. These surface maps are used to compute reflected fluxes in Chapter 3 and will be retrieved by the architecture described in Chapter 6, using said fluxes. We identify the following requirements for an algorithm that creates such surface maps:

- The algorithm should be efficient enough that a set of at least 10,000 different maps can be created.
- The patterns should be fractal in nature, roughly resembling those on Earth.
- It should be easy to control the fraction of the planet covered by a given surface type, so that this can be drawn from a distribution.

These requirements are met by first generating an elevation profile across the planet and then choosing an elevation range for the different surface types.

2.2.1. Elevation Profile

We investigated two different methods to create elevation maps:

- The Great Circle method⁶
- Tetrahedral Subdivision by Mogensen [30].

The Great Circle method works by creating random great circles on a sphere and raising the elevation of all points on one side of the great circle and lowering the elevation of all points on the opposite side. This method has the disadvantage that the maps are symmetrical, as elevations on opposite points of the sphere are always negative of each other. Although we did not investigate it, there seems to be a possibility that a neural network would become optimized to take advantage of this artefact, which is not representative of nature.

The Tetrahedral Subdivision method is described in the paper by Mogensen [30]. This method works by creating a random tetrahedron around the planet and then placing a new point on its longest edge (in the middle of the edge, with a random offset along the edge) to divide it into two new tetrahedra. This method is repeated until each facet on the sphere's surface is contained by a unique tetrahedron, with a unique elevation. By trial and error, we find that the offset exponent q discussed in the paper provides a roughly Earth-like level of map complexity for $q = 0.5$. An example elevation profile is shown in Figure 2.6.

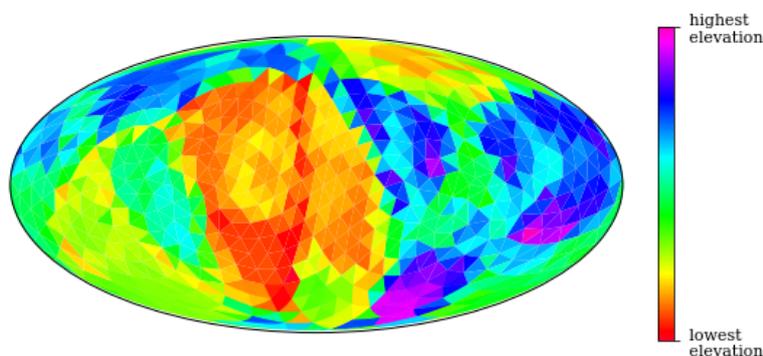


Figure 2.6: Example elevation profile created using the tetrahedral subdivision method developed by Mogensen [30]. A tetrahedron around the planet is subdivided repetitively until each facet is contained in a unique tetrahedron with a unique elevation. Each elevation range can be assigned a surface type to create planet maps. Recall that we use 1000 surface facets.

Due to the irregularity of the tetrahedra, patterns in the elevation profile are not easily detected, making this method advantageous compared to the Great Circle method. Another advantage is that the method is computationally more efficient. For these reasons, the Tetrahedral Subdivision method is chosen rather than the Great Circle method.

2.2.2. Surface Types

The next step to create the surface maps is to assign elevations ranges for each type of surface. For simplicity, only three types of surface are considered (in ascending order of elevation):

- Oceans are included as they are an indicator of habitability.
- Sandy desert is included as it is a common surface type on the terrestrial planets in our Solar System.
- Vegetation is included since finding evidence of life is the "holy grail" of exoplanet studies. The chlorophyll bump and red edge, shown in Figure 2.7, are two distinctive features of the reflection spectra of vegetation that can be used to identify it. It is likely that life on other planets has different absorption spectra than Earth's vegetation, but similar patterns could be observed (Fujii et al. [9], Seager et al. [35]).

To assign each facet on the planet a surface type, first a fraction of the total surface to be covered by each type should be defined. This is not a trivial problem as the following requirements should be met:

- Three numbers should be drawn from a random distribution.
- The numbers should be between 0 and 1.
- The numbers should sum to one.

⁶<http://paulbourke.net/fractals/noise/index.html>

- The numbers should each have equal covariances, so that a neural network cannot infer one surface type's presence due to another's presence.
- A significant portion of the numbers should be near 0 or 1, so that there are planets in the training data set that are nearly completely covered in a single surface type (such as Mars).

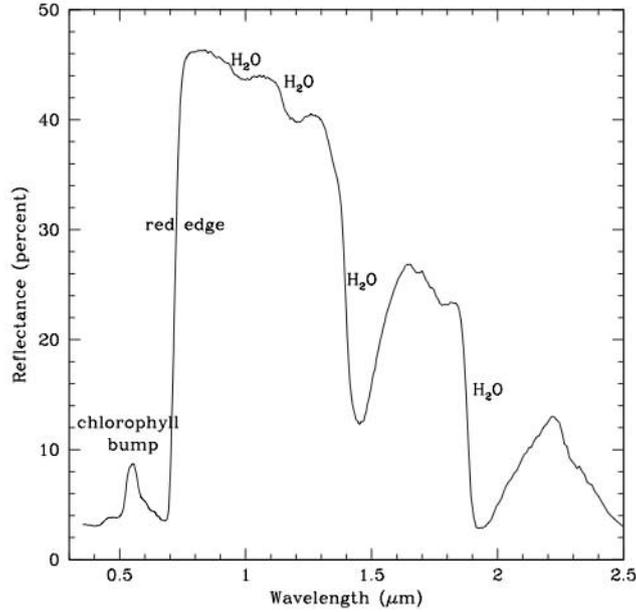


Figure 2.7: The reflection spectrum of a deciduous leaf shows the features of a chlorophyll bump and red edge (also shown in Figure 3.4), which are unique to plants on Earth. Vegetation is included as a surface type in the fake planet maps since similar features could be an indicator of life on an exoplanet. (Figure 1 from Seager et al. [35])

To fulfill these requirements, the following functions are used for a , b , and c , the fractions of surface covered by ocean, sandy desert and vegetation, respectively:⁷

$$\begin{aligned}
 x_1 &\sim \text{unif}(0, 1) \\
 x_2 &\sim \text{unif}(0, 1) \\
 a &= x_1^2 \\
 b &= (1 - a)(1 + \sin(\pi \cdot (x_2 - 0.5)))/2 \\
 c &= 1 - a - b
 \end{aligned} \tag{2.1}$$

such that the probability distribution for a , b and c is $f(x) = \frac{1}{2\sqrt{x}}$ on $[0,1]$. For each elevation map, a triplet of a , b , and c is created using the above equations. The elevation ranges that correspond to each surface type are then computed with ocean, sandy desert and vegetation in ascending order. Each facet is thus assigned a surface type depending on its elevation, creating maps such as the examples in Figure 2.8.

A set of 10,000 surface maps is created and to verify that the distributions in Equation 2.1 work as intended, each map's surface type fractions are plotted in a histogram in Figure 2.9. The covariance matrix of the number of surface type facets shows that the covariances are approximately equal:

$$\Sigma = \begin{bmatrix} 8.83 & -4.42 & -4.41 \\ -4.42 & 8.96 & -4.54 \\ -4.41 & -4.54 & 8.95 \end{bmatrix} \cdot 10^4 \tag{2.2}$$

This shows that no surface type is an indicator of another surface type's presence (off-diagonal values are the covariances of combinations of different surface types).

⁷Dr. Paul Visser came up with these equations.

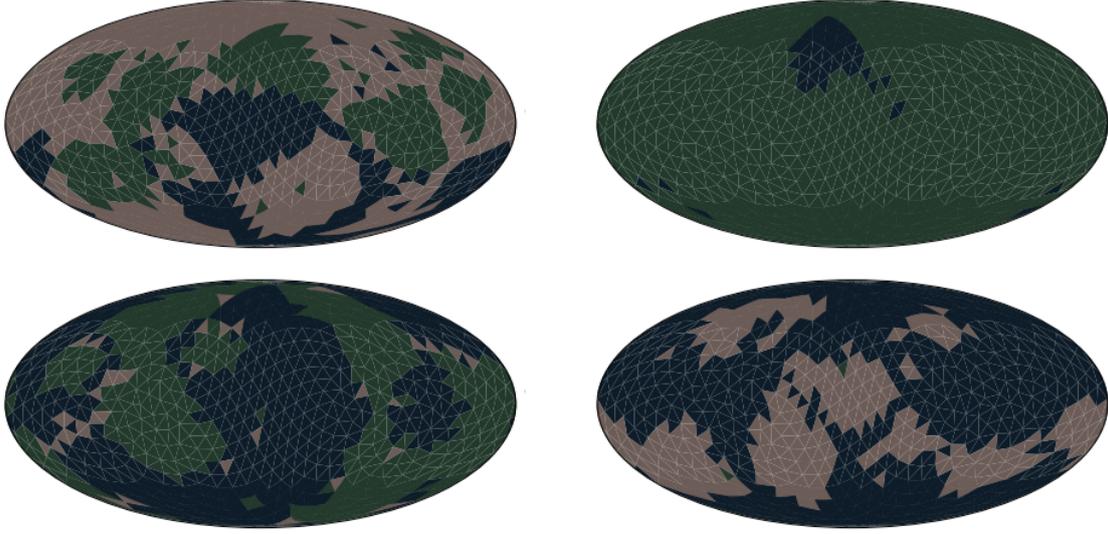


Figure 2.8: Examples picked from the set of 10,000 surface maps used for training the neural networks. The fraction of each surface type is drawn from the distributions in Equation 2.1 and used to determine each surface type's range on elevation maps like the one shown in Figure 2.6. Ocean, vegetation and desert are represented by dark blue, dark green and brown, respectively, with the RGB values being the effective albedos for the wavelengths 700, 550 and 500 nm, respectively (see Figure 3.4).

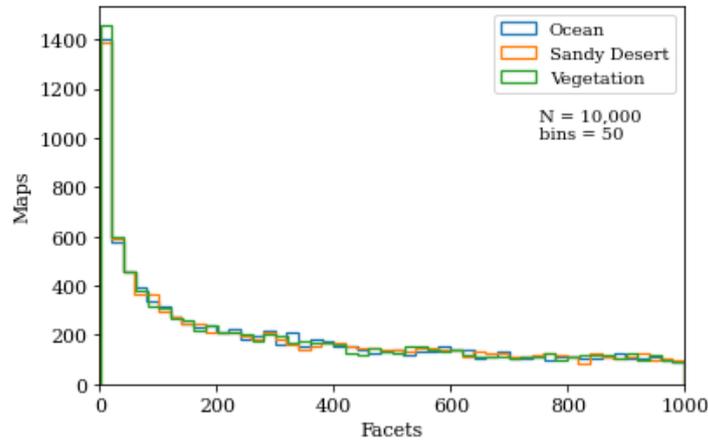


Figure 2.9: The number of facets of each surface type for all 10,000 surface maps. These are drawn from the distributions in Equation 2.1 following the distribution $\frac{1}{2\sqrt{x}}$. The covariances are roughly equal for each combination (see Equation 2.2). There are 10,000 surface type maps ($N = 10,000$) and 50 bins, which the number of facets are split into.

2.3. Cloud Maps

Like Earth and Venus, exoplanets may have clouds that hide the surface from the view of observers. These clouds can greatly effect the planetary climate and are thus of interest when assessing habitability (Karalidi et al. [20].) To simulate clouds patterns, the method from Rossi et al. [32] that creates patchy, zonal clouds is chosen. The planets' facets are projected into a geodetic (longitude-latitude) coordinate system, and then a facet on the planet is selected as the center of a cloud patch. Then 100 samples are drawn from a Gaussian distribution with a covariance matrix of:

$$\Sigma = \begin{bmatrix} 0.15 & 0 \\ 0 & 0.003 \end{bmatrix} [\text{rad}^2] \quad (2.3)$$

The asymmetry between longitude and latitude leads to zonal cloud patterns, similar to those observed on Earth and other planets such as Jupiter. This process can be repeated until the desired cloud coverage is achieved, which is drawn from the first distribution in Equation 2.1, independently of the surface cover draws. A set of 10,000 cloud patterns is created, of which four examples are shown in Figure 2.10. The cloud

covers of the maps are plotted in a histogram in Figure 2.11, to verify that the distributions in Equation 2.1 work correctly again.

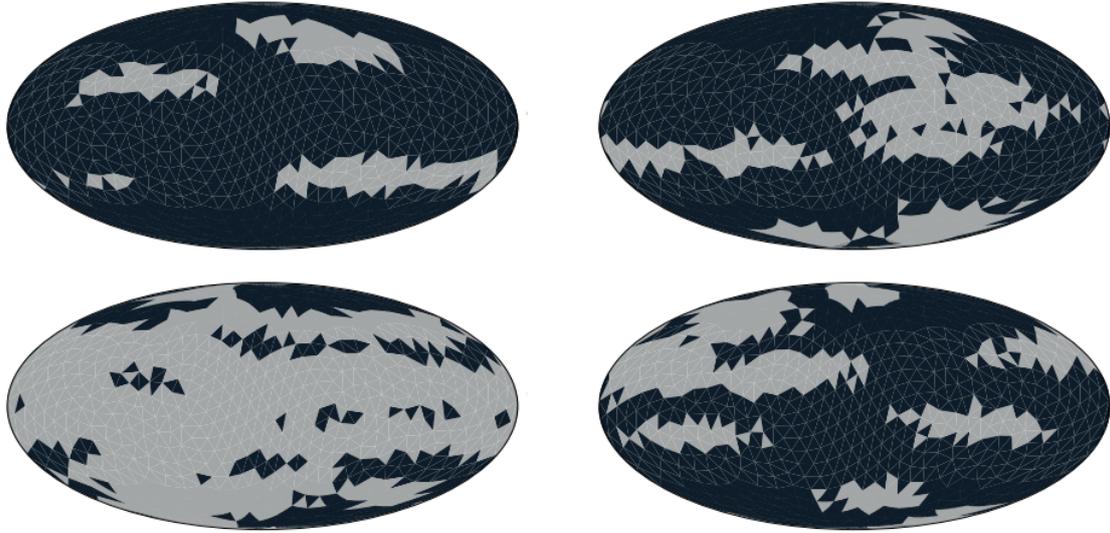


Figure 2.10: Examples picked from the set of 10,000 cloud maps, created by picking random locations on the map as a center of a cloud and making 100 draws from a Gaussian distribution until the desired cloud cover is reached. The longitudinal width is on average $\sqrt{50} \approx 7$ times greater than the latitudinal height, due to the asymmetry in the covariance matrix in Equation 2.3.

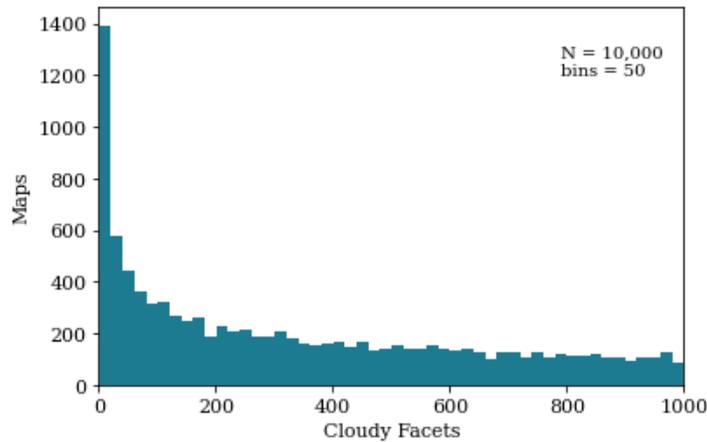


Figure 2.11: The number of cloudy facets for each of the 10,000 cloud maps, drawn from the first distribution in Equation 2.1.

2.4. Inclinations

The inclination, i , of a planet's orbit is defined as the angle between the orbital plane and the reference plane containing the observer, as shown in Figure 2.12. This means that an orbit that is seen on its side (or edge-on) has an inclination of 0° and a face-on orbit has an inclination of 90° .⁸ This angle is important for retrievals as it determines the range of phase angles, α , (discussed in Chapter 3) that are visible to the observer to $i \leq \alpha \leq 180^\circ - \alpha$. For simplicity, the studied inclinations are limited to steps of 15° between 0° and 90° :

$$i \in \{0, 15, 30, 45, 60, 75, 90\} \text{ [deg]} \quad (2.4)$$

Inclinations outside of $0^\circ \leq i \leq 90^\circ$ are redundant since these orbits are mirror images of ones inside these inclinations.

⁸Note that in many exoplanet papers the opposite is true.

⁹Image Source: <https://commons.wikimedia.org/wiki/File:Orbit1.svg#/media/File:Orbit1.svg>

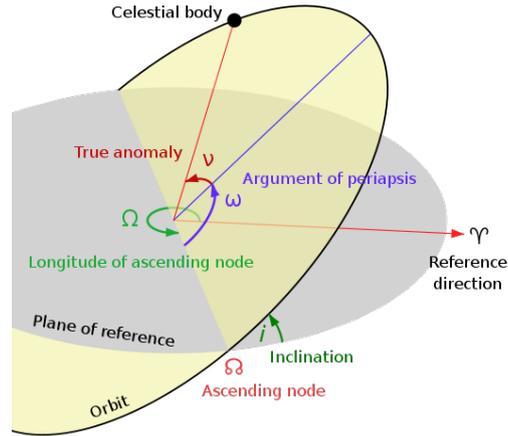


Figure 2.12: Orbital Elements. In this report Υ is defined as the direction toward the observer and Ω is always $\pi/2$. Other Ω values are redundant as they are equivalent to a rotation of the reference frame.⁹

2.5. Rotation Axes

An exoplanet's rotational period and axis of rotation modulate the observed light curves and must therefore be retrieved to map an exoplanet's surface (Kawahara [22].) The rotation axes of the planets in our Solar System exhibit a bias towards a small axial tilt, but this need not be the case for all exoplanets, as large impacts and/or planetary migration can cause a change in rotation axis, which is believed to be the reason that Uranus has an axial tilt of 98° (Lissauer and Pater [28].)

So that the neural networks do not learn a bias towards planets with a small axial tilt, we decide to use rotation axes that are homogeneously distributed across a unit sphere's surface. Additionally, the neural networks should not be able to resolve degeneracies by learning the sampling of rotation axes. A degeneracy occurs when two unique planets create the same light curves, which we find to be the case for two planets in side-on orbits, whose maps are flipped about the equator and whose rotation axes are the same in the out-of-orbital plane direction and negatives of each other in the orbital plane. This degeneracy is discussed in more detail in Section 4.2. So that the neural networks cannot resolve this degeneracy using the sampling, each rotation axis' degenerate counterpart is also be included in the set. To achieve this, the first 32 points from a 64-point Fibonacci spiral (see Section 2.1.2) as well as their degenerate counterparts (found by multiplying element-wise by $[-1, -1, 1]$) create a set of 64 axes, which can be seen in Figure 2.13.

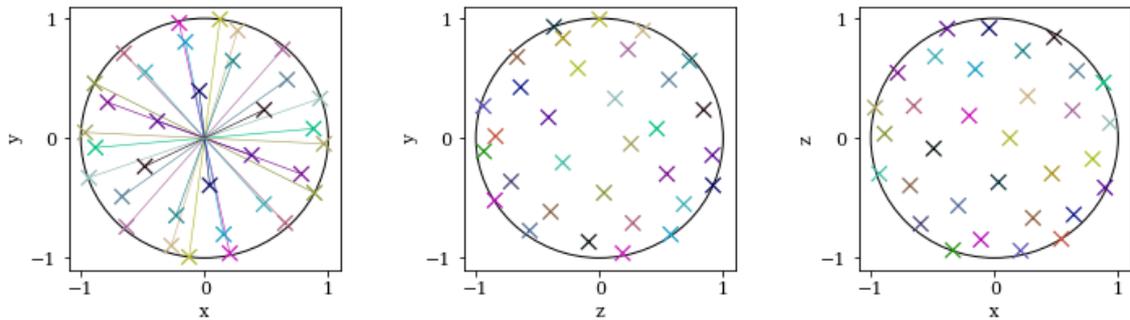


Figure 2.13: Locations of the 64 rotation axes projected onto xy (left), zy (middle) and xz (right) planes. The rotation axes are taken from the first 32 values of a Fibonacci spiral sampled at 64 locations and their degenerate counterparts (see Section 4.2). Degenerate counterparts are plotted in the same color and connected by a line. Only half of the axes on the 'visible' side of the unit sphere are plotted.

2.6. Combinations

In Section 2.2 - Section 2.5 the set of possibilities for each of the four planetary parameters are defined. The set size for each is shown in Table 2.1, resulting in a total of 44.8 billion unique combinations. To create a training data set with 4 million planets, each parameter is drawn from its set and the four parameters are

stored as four integer indices. The first four planets in the data set are shown in Figure 2.14.

Surface Maps	10,000
Cloud Maps	10,000
Rotation Axes	64
Inclinations	7

Table 2.1: The set size for each of the four planetary parameters. Multiplication results in a total of 44.8 billion unique combinations.

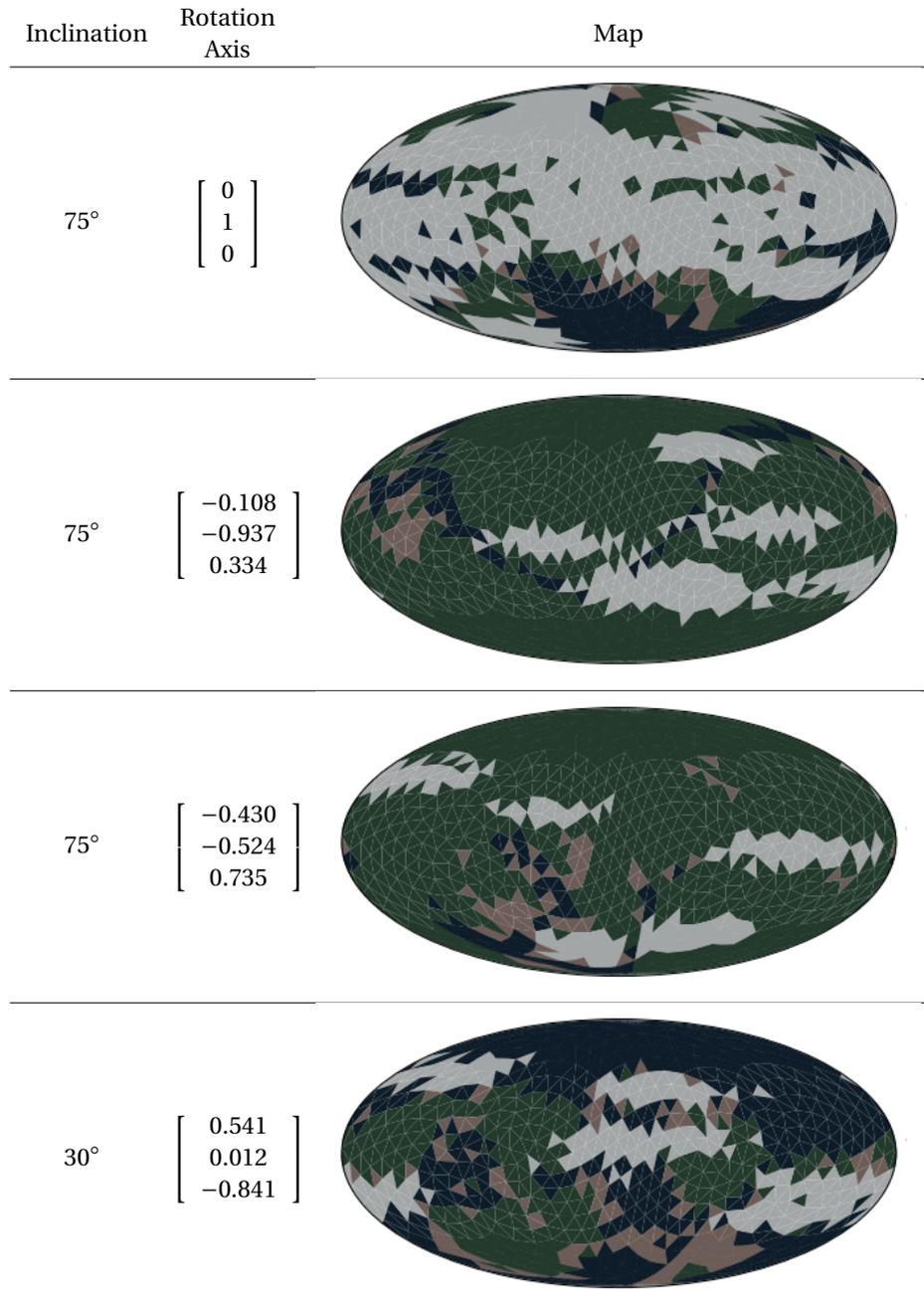


Figure 2.14: The first four planets in the training data set of 4 million. The fluxes of each planet are calculated as described in Chapter 3 and used to train the neural networks. Dark blue, brown, green and gray represent ocean, sandy desert, vegetation and clouds, respectively. The inclination is defined such that side-on and face-on orbits have inclinations of 0° and 90°, respectively. The observer is in the direction of the x-axis.

3

Computing Reflected Fluxes

The next step for the training of the neural networks is computing the reflected light signals of the planets. This thesis differs from the approach used by others (see for example Fujii and Kawahara [8], Kawahara and Masuda [23], Fan et al. [6], Farr et al. [7], Asensio Ramos and Pallé [2]) in several aspects.

One improvement is that the method developed by Rossi et al. [32] and Groot et al. [12] is used to calculate fluxes, allowing for directional dependence of the reflected light (Section 3.1). Effects such as ocean glint and rainbows are also included, causing significant deviations from Lambertian reflection signals used by other authors. Another advantage of this approach is that the polarization of the reflected light is also calculated, which has been shown by Trees and Stam [39] and Groot et al. [12] to carry information about oceans and clouds and could thus be used to increase mapping accuracy.

A second deviation, discussed in Section 3.2, is that rather than continuous measurements throughout one complete orbit, the planet is sampled at 8 locations in its orbit. The locations are selected such that the phase angle (defined in Figure 3.1) is in the range $38^\circ \leq \alpha \leq 142^\circ$ since observation becomes impossible if the planet is too close to its star. At each location the planet is frozen in position and rotated about its axis for 8 observations (Section 3.3). When compared to using continuous measurements throughout a complete orbit, this approach greatly reduces memory requirements for storing the light curves, simplifies the network architectures and increases neuron gradients, leading to more efficient training.



Figure 3.1: The planetary phase angle α .¹⁰

Another difference compared to work by other authors is the modelling of noise (Section 3.5). Rather than using Gaussian noise, which creates negatives values for some observations, the more accurate model of shot noise is used. Normalization of the flux curves is also included so that only relative fluxes and not absolute fluxes are used by the retrieval algorithms, since the radius of an exoplanet is impossible to constrain using direct observations (Section 3.6).

By simulating the light curves of a model Earth (Figure 3.9), we show that Stokes parameter U is of negligible magnitude and would thus be dominated by noise. Hence, it is not used for the retrievals in our neural networks.

¹⁰Images of the Earth and Nancy Grace Roman Space Telescope taken from https://commons.wikimedia.org/wiki/File:The_Earth_seen_from_Apollo_17_with_transparent_background.png and https://www.nasa.gov/sites/default/files/wfirst-afta_0.jpg, respectively.

3.1. Reflection Models

Two models are used to simulate the reflection from a planet in this thesis:

- Lambertian reflection is used since it is the standard model used by other authors in the field of exocartography and so is useful for comparison and verification.
- Directional reflection using Fourier files as described by Rossi et al. [32]. This method allows for more accurate modelling of flux contributions, including ocean glint, rainbows and and polarized fluxes.

Both reflection models require that the three vectors in Figure 3.2 are defined for each of the planet's facets: \mathbf{v}_{norm} is the vector normal to the facet surface, which can be computed by taking the cross product of two edges, \mathbf{v}_{obs} is the vector towards the observer and is defined to always be $[1,0,0]$ and \mathbf{v}_{star} is the vector towards the star, which is found using the methods described in Section 3.2. All three are unit vectors. Using these vectors, the flux of a planet can be calculated by summing the contributions of its 1000 facets.

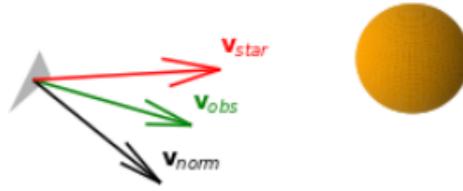


Figure 3.2: The three vectors used to determine the flux contribution of a facet: the vector to the star, the vector to the observer and the normal vector. All three are unit vectors.

3.1.1. Directional Reflection

To accurately simulate fluxes and include polarization, the method described in Rossi et al. [32] is used. This uses radiative transfer computations with a model atmosphere consisting of homogeneous layers containing cloud, aerosol and/or gas particles. The atmosphere used approximates Earth's, with a depolarization factor (Hansen and Travis [13]) of 0.0279, a surface pressure of 1.013 bar and 16 stacked atmospheric layers. The outputs of the radiative transfer code are Fourier coefficients, R , that can be used to calculate the flux using the following equations, from Rossi et al. [32]:

$$\begin{aligned}
 I(\mu, \mu_0, \phi - \phi_0, \lambda) / \mu_0 F_0(\lambda) &= R_{11}^0(\mu, \mu_0, \lambda) + 2 \sum_{m=1}^M \cos m(\phi - \phi_0) R_{11}^m(\mu, \mu_0, \lambda) \\
 Q(\mu, \mu_0, \phi - \phi_0, \lambda) / \mu_0 F_0(\lambda) &= R_{21}^0(\mu, \mu_0, \lambda) + 2 \sum_{m=1}^M \cos m(\phi - \phi_0) R_{21}^m(\mu, \mu_0, \lambda) \\
 U(\mu, \mu_0, \phi - \phi_0, \lambda) / \mu_0 F_0(\lambda) &= 2 \sum_{m=1}^M \sin m(\phi - \phi_0) R_{31}^m(\mu, \mu_0, \lambda)
 \end{aligned} \tag{3.1}$$

with F_0 being the incident flux (we assume $F_0 = 1$ per unit area) and R_{11}^m , R_{21}^m and R_{31}^m being the m th Fourier coefficients for I , Q and U respectively. The files contain Fourier coefficients for 21 x 21 and 101 x 101 points of μ and μ_0 , respectively, where:

$$\mu = \mathbf{v}_{norm} \cdot \mathbf{v}_{star}, \quad \mu_0 = \mathbf{v}_{norm} \cdot \mathbf{v}_{obs} \tag{3.2}$$

Since the coefficients are discrete points in the continuous $\mu\mu_0$ plane, each facet's R is found by nearest neighbor interpolation, which is far more efficient than bilinear or bicubic interpolation (roughly 30 times as efficient when using the SciPy Python package).¹¹ To further increase efficiency, interpolation and subsequent calculations are only necessary when $\mu > 0$ and $\mu_0 > 0$ as otherwise the flux equals zero. The azimuthal difference angle, $\phi - \phi_0$, is found by projecting \mathbf{v}_{obs} and \mathbf{v}_{star} onto the facet, finding their determinant and dot product, and then passing both into the arctan2 function, respectively. The variables μ , μ_0 and $\phi - \phi_0$ are illustrated in Figure 3.3 and are used with Equation 3.1 to calculate I , Q and U that together define the Stokes vector:

¹¹<https://docs.scipy.org/doc/scipy/reference/generated/scipy.interpolate.griddata.html>

$$\mathbf{I} = \begin{bmatrix} I \\ Q \\ U \end{bmatrix} \quad (3.3)$$

where I is the total flux and Q and U are perpendicularly polarized linear fluxes. The circularly polarized flux, V , is excluded from the Stokes vector since Groot et al. [12] show that it is approximately 5 orders of magnitude smaller than I for an Earth-like planet and thus detection is nearly impossible, as noise dominates the signal.

For sandy desert and vegetation, only 3 Fourier coefficients are used since Lambertian reflection of the surface is assumed, while for the ocean 90 are used. The resulting fluxes are defined with respect to the local meridian plane and must be multiplied by the rotation matrix, \mathbf{L} , (Equation 2 in Rossi et al. [32]) to translate into the planetary scattering plane before summing the facet contributions:

$$\mathbf{L}(\beta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 2\beta & \sin 2\beta \\ 0 & -\sin 2\beta & \cos 2\beta \end{bmatrix} \quad (3.4)$$

where β (also shown in Figure 3.3) is the angle between the local meridian plane and the planetary scattering plane, calculated by taking the clockwise angle from \mathbf{v}_{norm} to \mathbf{v}_{star} when both are projected into a plane normal to the observer.

The sandy desert and vegetation surfaces are modelled as Lambertian reflectors and the output files are provided by Dr. Daphne Stam. The files for the ocean are provided by Victor Trees, with the ocean modelled as a rough Fresnel reflecting surface with wave facet inclinations using a wind speed of 7 m/s (Trees and Stam [39]).

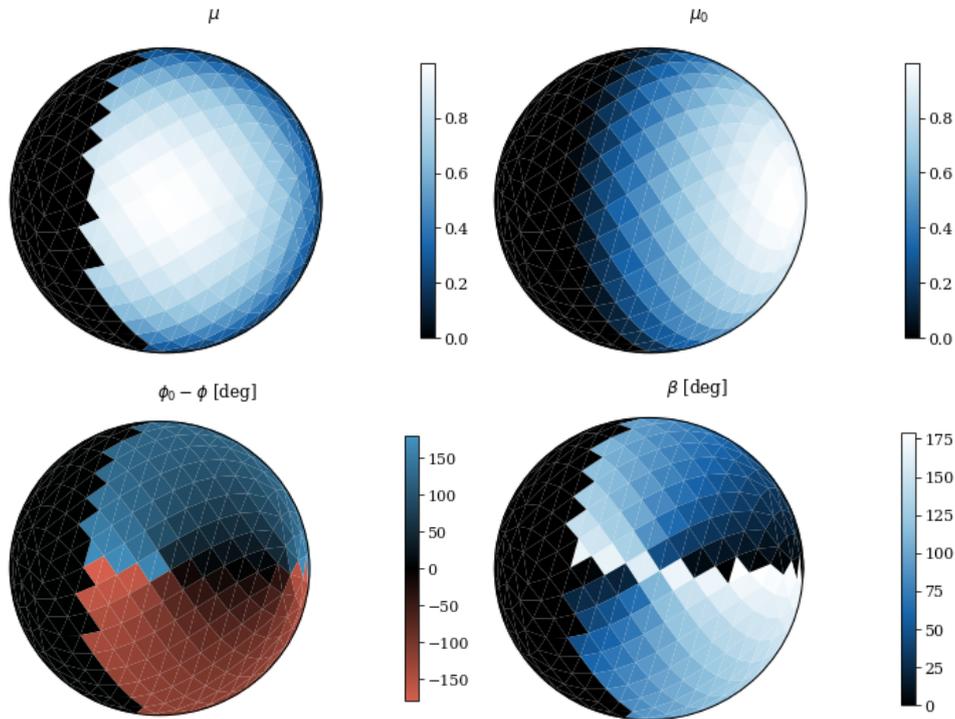


Figure 3.3: Variables used to calculate the directional reflection using Equation 3.1. This planet's star is to the right at a phase angle of 60° .

Since the number of coefficients in the cloud files is too large to transfer easily, Dr. Daphne Stam instead calculated the reflected Stokes vector for a grid of 100×100 pixels on the planetary disc of a cloud covered planet for phase angles of:

$$\alpha \in \{0, 20, 30, 35, 38, 40, 42, 45, 50, 60, 80, 90, 100, 120, 140, 160, 180\} \text{ [deg]} \quad (3.5)$$

The flux of a cloudy facet can thus be found by selecting the nearest pixel on the planetary disc, and linearly interpolating between phase angles. In these files the star is taken always to be to the right of the planet and so the position on the planet needs to be translated between reference frames. The inhomogeneous distribution of phase angles ensures that the flux is accurate near phase angles with large flux gradients like the rainbow feature at $\alpha = 38^\circ$.

3.1.2. Lambertian Reflection

The Lambertian flux of a facet is calculated using its albedo, area and the three vectors in Figure 3.2:

$$I = \text{albedo} \cdot \text{area} \cdot \max(0, \mu) \cdot \max(0, \mu_0) \quad (3.6)$$

The maximum functions are included in the equations so that facets facing away from the star or observer contribute zero flux, rather than negative flux. The equation can be simplified by using a \mathbf{v}_{norm} that is equal in magnitude to the area of the facet when computing μ or μ_0 . This is easily achieved by computing \mathbf{v}_{norm} as half the cross product of two of the triangle's edges. The total flux of a planet can then be computed by summing the contributions of all 1000 facets. The calculation is verified by confirming that the flux of a completely white planet with a radius of 1 and at full phase is equal to $2/3 \cdot \pi$ (Stam et al. [37]). A deviation of 0.6% is found due to the facet approximation of a sphere.

For ease of comparison between the directional and Lambertian reflection models, the effective albedos of each surface type are defined such that at full phase a completely homogeneous planet has the same flux using either model. Each surface type's effective albedo is therefore computed by calculating the flux of a homogeneous planet with a radius of 1 at full phase using directional reflection and dividing by $2/3 \cdot \pi$. These results, shown in Figure 3.4, differ from the surface albedos due to the Rayleigh scattering contribution to the planetary flux, which is largest for the smallest wavelengths. This gives Earth its characteristic blue tint that can be seen in the Pale Blue Dot picture.

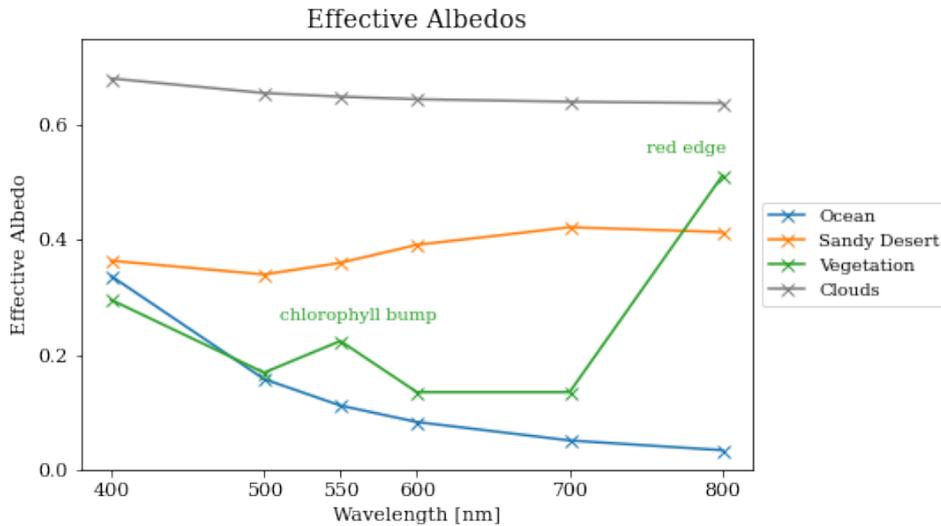


Figure 3.4: The effective albedos of sandy desert, vegetation, clouds and ocean, found by computing the planetary flux of a homogeneous planet, including atmospheric scattering, at full phase. The 550 nm wavelength is included so that the chlorophyll bump can be resolved. The albedos at 700, 550 and 500 nm are used for the RGB color values respectively for the maps in this thesis.

3.1.3. Phase Curve Comparison

The results are first verified by computing phase curves of homogeneous planets, shown in Figure 3.5, and comparing to Figures 5 and 7 from Trees and Stam [39] for ocean and clouds, respectively, and Figure 4 from Stam [36] for vegetation and sandy desert. The normalization in these papers is such that a Lambertian white disc at full phase gives a flux of 1 while in this thesis it is such that a Lambertian white sphere at full phase gives a flux of 1, so the results in Figure 3.5 must be divided by 1.5 for comparison.

For nearly all phase curves there is a significant deviation between the Lambertian and directional phase curves. The exceptions to this are vegetation and desert-covered surfaces, which are modelled as Lambertian

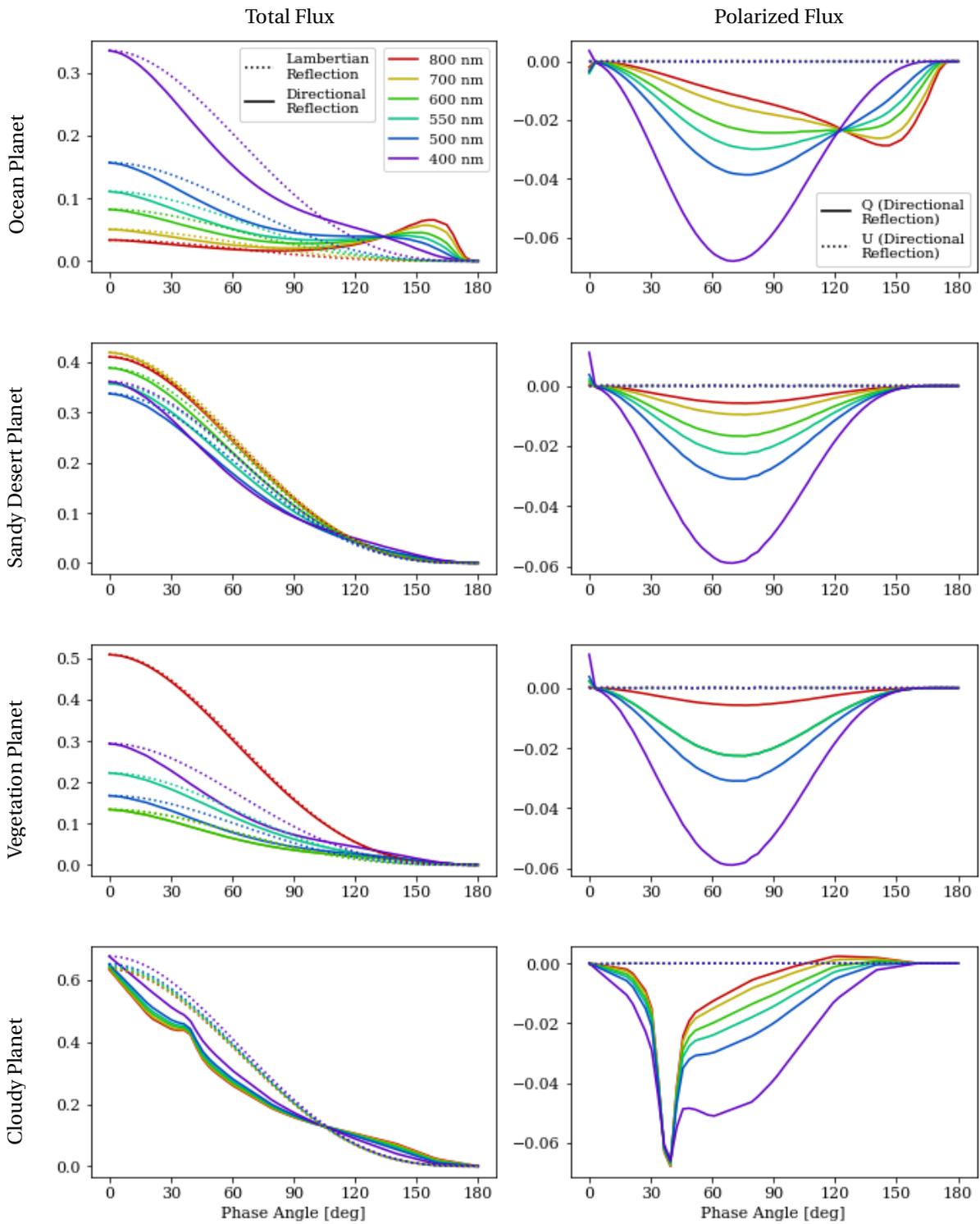


Figure 3.5: Phase curves for homogeneous planets of different surface types. The phase angle, α , is defined in Figure 3.1. These curves are verified by comparing to Figures 5 and 7 from Trees and Stam [39] for ocean and clouds, respectively, and Figure 4 from Stam [36] for vegetation and sandy desert. The results should be divided by 1.5 for comparison due to differences in normalization. The polarized fluxes Q and U are computed using only directional reflection, since this is not possible with the Lambertian model. At $\alpha = 0^\circ$ there is a numerical artefact that causes the polarized flux to diverge from 0 since the vectors to the star and observer are equal.

surfaces. For wavelengths greater than 600 nm, atmospheric Rayleigh scattering is negligible compared to surface reflection so the directional and Lambertian light curves are nearly identical. The effect of Rayleigh scattering increases for smaller wavelengths and hence the deviation from the Lambertian phase curve is largest at 400 nm for vegetation and sandy desert. The Lambertian assumption overestimates the flux at phase angles below 90° , and underestimates it for phase angles above 120° , due to forward scattering in the atmosphere at the edges of the planetary disk. This means that retrieval algorithms relying on the Lambertian model of reflection overestimate the albedo of a planet at low phase angles and underestimate it at large phase angles due to atmospheric effects, even when the underlying surfaces are Lambertian.

The differences are even more pronounced for the cloudy planet, due to a relatively large negative gradient at full phase, leading to a great overestimation of the flux for phase angles below 90 degrees. Due to forward scattering, the Lambertian flux is also particularly underestimated when clouds are present at high phase angles. This forward scattering increases in magnitude for higher wavelengths and so the greatest difference is for a wavelength of 800 nm. At a phase angle of 38 degrees, there is also a slight flux increase due to the rainbow effect of the cloud particles, although this effect is primarily seen in the linearly polarized flux Q . No other surface type has such a large linearly polarized flux at low phase angles and high wavelengths, which can be used as an indicator of clouds by a retrieval algorithm.

Finally, the surface type with the largest deviation from Lambertian flux curves is the ocean planet since it exhibits a much higher flux for high phase angles due to ocean glint. This effect increases with increasing wavelength as Rayleigh scattering becomes weaker. Consequently, increasing intensity with increasing phase at a wavelength of 800 nm can be used as an indicator of an ocean, as shown by Trees and Stam [39]. This is also true for the polarized flux of an ocean planet, which is significantly higher than the other studied surface types for high phase angles, especially for high wavelengths.

3.1.4. Resolved Flux Comparison

To further illustrate the differences between Lambertian and directional reflection, resolved fluxes on the planetary surface are shown in Figure 3.6. Only cloud planets and ocean planets are shown, as these are the most interesting and suffice for the ensuing discussion. For the ocean planet, the glint pattern can be seen most clearly for high phase angles, as expected from the phase curves in Figure 3.5. At $\alpha = 142^\circ$ and $\lambda = 800$ nm, the glint is very bright with a magnitude of 2.7, meaning that the underlying surface can be unequivocally classified as ocean when compared to clouds, vegetation or sandy desert, if resolved. As the phase angle decreases, the glint weakens and thus is less predominant in the planetary signal. For $\lambda = 400$ nm the glint pattern is much weaker, reaching a maximum of roughly 0.8, since Rayleigh scattering dominates the planetary signal and prevents light that is reflected by the ocean from reaching the observer.

An interesting effect of Rayleigh scattering is that the flux of the subsolar facets is not as high as predicted by Lambertian reflection. This can be seen for both ocean and clouds at phase angles of 90° and 142° . Thus, a Lambertian retrieval algorithm is overly optimistic about the concentration of the planetary signal to a specific region. For example, for the cloudy planet at $\alpha = 90^\circ$, the Lambertian assumption would assume that nearly all light being received is from the sub-solar region, while the directional model shows that the flux contributions are more evenly distributed. This can also be seen for sandy desert and vegetation planets (not shown).

As seen in Figure 3.5, the Lambertian model greatly underestimates the flux of the cloudy planet for high phase angles, where forward scattering of light is predominant in the signal. Even a small number of cloudy facets could lead a retrieval algorithm that assumes Lambertian reflection to greatly overestimate the albedo of this region.

3.2. Orbit

To determine the directions towards the star and the observer from each facet on the planet, first the location of the planet needs to be calculated. As described in Section 2.4, the orbital inclinations are drawn from a set ranging from 0° to 90° in steps of 15° . Since changing the orbital eccentricity and radius alters only the magnitudes of the light curves, only circular orbits with a radius of 1 are considered for simplicity.

In the coordinate system used to simulate the orbit, the x-axis points to the observer, the ascending node of the orbit is on the y-axis and the z-axis completes the Cartesian coordinate system, as shown in Figure 3.7. Edge-on orbits are thus in the xy plane and face-on orbits are in the yz plane. The angle of rotation of the exoplanet around its orbit is defined as u , starting at the intersection with the positive y-axis (each orbit intersects the y-axis at $[0,1,0]$ and $[0,-1,0]$, regardless of inclination). Thus the position of the exoplanet in its

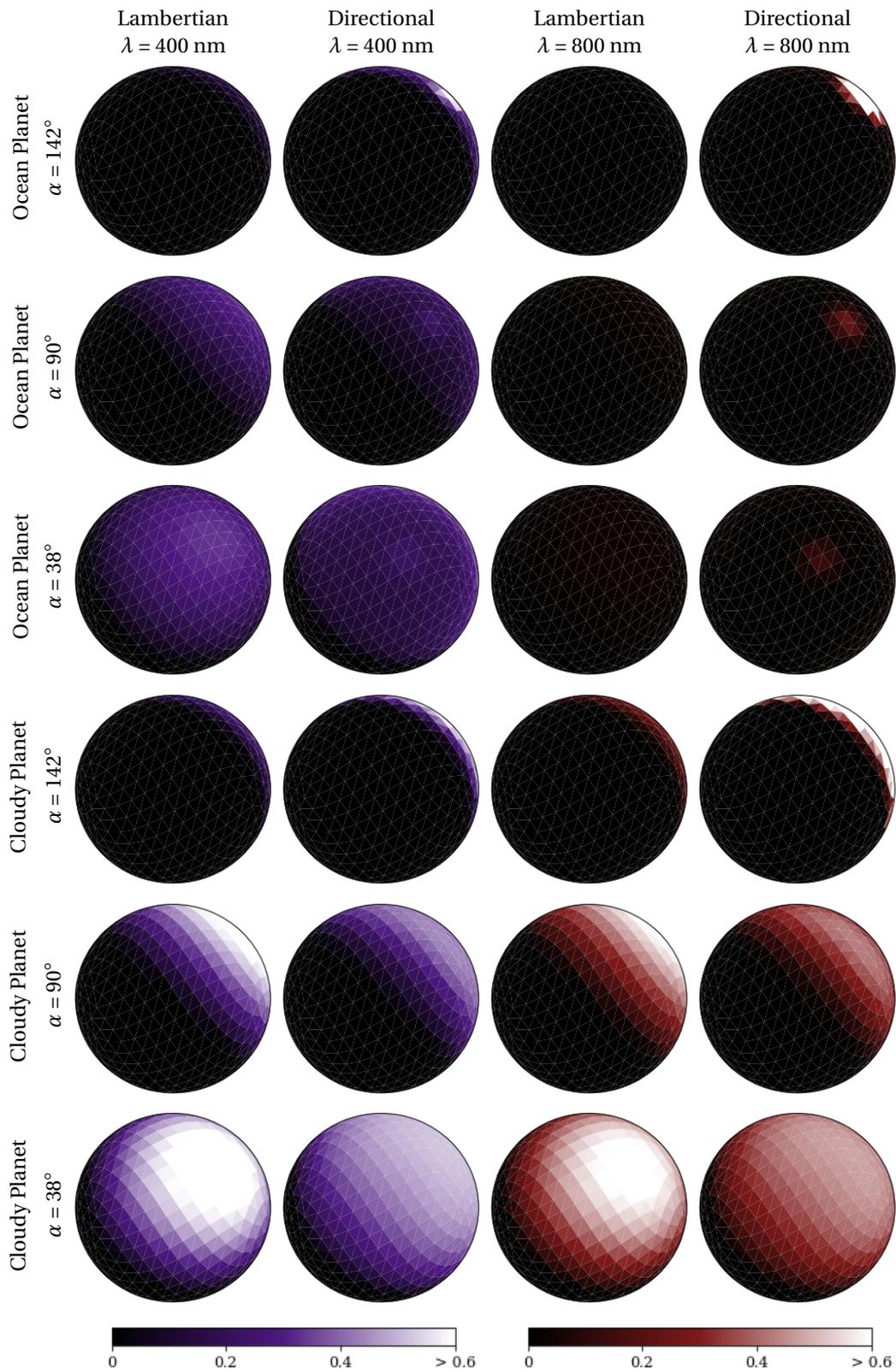


Figure 3.6: Resolved fluxes with the star to the top-right of the planet. A brightness of 1 corresponds to a completely white Lambertian surface with normally incident star light. The most extreme value is 2.7 for ocean glint at $\alpha = 142^\circ$ and $\lambda = 800 \text{ nm}$. $\alpha = 142^\circ$ and $\alpha = 38^\circ$ are shown here since these are the closest positions to the star that are considered possible in Section 3.2. The Lambertian assumption greatly underestimates the fluxes of cloudy planets and ocean-covered planets at high phase angles. At lower phase angles the Lambertian assumption overestimates the concentration of the flux to the sub-solar point.

orbit is given by:

$$\text{position} = \begin{bmatrix} \cos u \\ \cos i \cdot \sin u \\ \sin i \cdot \sin u \end{bmatrix} \quad (3.7)$$

The angle of rotation in the yz plane is defined as γ , in anti-clockwise direction from the observer's perspective, starting at the intersection with the positive y -axis.

For each inclination, the exoplanet is observed at 8 locations in the orbit. The range of possible phase angles throughout the orbit is $i \leq \alpha \leq 180^\circ - i$. Only locations inside the phase angle range of $38^\circ \leq \alpha \leq 142^\circ$ are considered since exoplanets cannot be directly observed when too close to their parent star. 38° is selected as the inner phase angle so that the rainbow is visible, which could be recommended as a requirement for future telescopes to help identify possible water clouds (Karalidi et al. [20]). Orbits with inclinations above 38° do not intersect the rainbow and the observation epochs are evenly distributed across the range $0^\circ \leq \gamma < 360^\circ$, to maximize modulation across the planet's surface. For inclinations below 38° , four observations are made at the inner most points possible, 2 at $\alpha = 38^\circ$ and $\alpha = 142^\circ$ each. The remaining four locations are distributed evenly within the range of possible γ . For an edge-on orbit, γ is either 0° or 180° , so in this case the values of u that the planet is observed at are found by taking the limit as $i \rightarrow 0^\circ$.

The resulting values of u , α and γ for each epoch are presented in Table 3.1 and plotted from the observer's perspective in Figure 3.7.

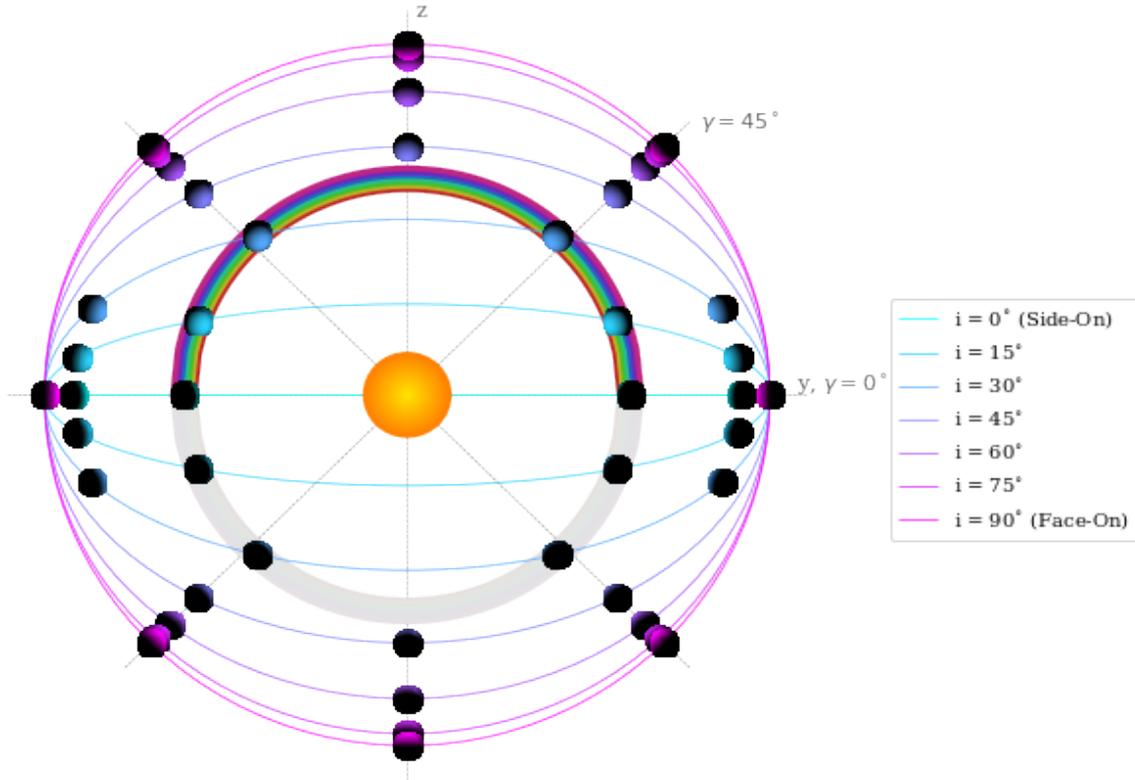


Figure 3.7: Observation locations from the observer's perspective. The rainbow is located at $\alpha = 38^\circ$ and is reversed from a traditional rainbow due to the small size of the cloud water droplets compared to rain droplets. Since $\alpha \geq 90^\circ$ for all positions below the star in the figure, the rainbow's "reflection" is at $\alpha = 142^\circ$. All points with the same $\alpha = x$ and $180^\circ - \alpha$ lie on circle from the observer's perspective for the same reason a traditional rainbow appears circular, with the positions of the light source and observer being flipped. The light-gray dotted lines show γ in eight evenly distributed steps of 45° . The right and left-most planets for $i = 90^\circ$ are covering those for $i \in \{45^\circ, 60^\circ, 75^\circ\}$ and for the edge-on orbit ($i = 0^\circ$) the planets with $\alpha > 90^\circ$ are covering those with $\alpha < 90^\circ$. The numerical values of u , α and γ at each epoch can be found in Table 3.1.

Incl. i	Epoch 1	Epoch 2	Epoch 3	Epoch 4	Epoch 5	Epoch 6	Epoch 7	Epoch 8
0°	$u = 23^\circ$	$u = 52^\circ$	$u = 128^\circ$	$u = 157^\circ$	$u = 203^\circ$	$u = 232^\circ$	$u = 308^\circ$	$u = 337^\circ$
	$\alpha = 67^\circ$	$\alpha = 38^\circ$ 🌈	$\alpha = 38^\circ$ 🌈	$\alpha = 67^\circ$	$\alpha = 113^\circ$	$\alpha = 142^\circ$	$\alpha = 142^\circ$	$\alpha = 113^\circ$
	$\gamma = 0^\circ$	$\gamma = 0^\circ$	$\gamma = 180^\circ$	$\gamma = 180^\circ$	$\gamma = 180^\circ$	$\gamma = 180^\circ$	$\gamma = 0^\circ$	$\gamma = 0^\circ$
15°	$u = 24^\circ$	$u = 55^\circ$	$u = 125^\circ$	$u = 156^\circ$	$u = 204^\circ$	$u = 235^\circ$	$u = 305^\circ$	$u = 336^\circ$
	$\alpha = 67^\circ$	$\alpha = 38^\circ$ 🌈	$\alpha = 38^\circ$ 🌈	$\alpha = 67^\circ$	$\alpha = 113^\circ$	$\alpha = 142^\circ$	$\alpha = 142^\circ$	$\alpha = 113^\circ$
	$\gamma = 7^\circ$	$\gamma = 20^\circ$	$\gamma = 160^\circ$	$\gamma = 173^\circ$	$\gamma = 187^\circ$	$\gamma = 200^\circ$	$\gamma = 340^\circ$	$\gamma = 353^\circ$
30°	$u = 30^\circ$	$u = 66^\circ$	$u = 114^\circ$	$u = 150^\circ$	$u = 210^\circ$	$u = 246^\circ$	$u = 294^\circ$	$u = 330^\circ$
	$\alpha = 65^\circ$	$\alpha = 38^\circ$ 🌈	$\alpha = 38^\circ$ 🌈	$\alpha = 65^\circ$	$\alpha = 115^\circ$	$\alpha = 142^\circ$	$\alpha = 142^\circ$	$\alpha = 115^\circ$
	$\gamma = 16^\circ$	$\gamma = 48^\circ$	$\gamma = 132^\circ$	$\gamma = 164^\circ$	$\gamma = 196^\circ$	$\gamma = 228^\circ$	$\gamma = 312^\circ$	$\gamma = 344^\circ$
45°	$u = 0^\circ$	$u = 55^\circ$	$u = 90^\circ$	$u = 125^\circ$	$u = 180^\circ$	$u = 235^\circ$	$u = 270^\circ$	$u = 305^\circ$
	$\alpha = 90^\circ$	$\alpha = 55^\circ$	$\alpha = 45^\circ$	$\alpha = 55^\circ$	$\alpha = 90^\circ$	$\alpha = 125^\circ$	$\alpha = 135^\circ$	$\alpha = 125^\circ$
	$\gamma = 0^\circ$	$\gamma = 45^\circ$	$\gamma = 90^\circ$	$\gamma = 135^\circ$	$\gamma = 180^\circ$	$\gamma = 225^\circ$	$\gamma = 270^\circ$	$\gamma = 315^\circ$
60°	$u = 0^\circ$	$u = 49^\circ$	$u = 90^\circ$	$u = 131^\circ$	$u = 180^\circ$	$u = 229^\circ$	$u = 270^\circ$	$u = 311^\circ$
	$\alpha = 90^\circ$	$\alpha = 68^\circ$	$\alpha = 60^\circ$	$\alpha = 68^\circ$	$\alpha = 90^\circ$	$\alpha = 112^\circ$	$\alpha = 120^\circ$	$\alpha = 112^\circ$
	$\gamma = 0^\circ$	$\gamma = 45^\circ$	$\gamma = 90^\circ$	$\gamma = 135^\circ$	$\gamma = 180^\circ$	$\gamma = 225^\circ$	$\gamma = 270^\circ$	$\gamma = 315^\circ$
75°	$u = 0^\circ$	$u = 46^\circ$	$u = 90^\circ$	$u = 134^\circ$	$u = 180^\circ$	$u = 226^\circ$	$u = 270^\circ$	$u = 314^\circ$
	$\alpha = 90^\circ$	$\alpha = 79^\circ$	$\alpha = 75^\circ$	$\alpha = 79^\circ$	$\alpha = 90^\circ$	$\alpha = 101^\circ$	$\alpha = 105^\circ$	$\alpha = 101^\circ$
	$\gamma = 0^\circ$	$\gamma = 45^\circ$	$\gamma = 90^\circ$	$\gamma = 135^\circ$	$\gamma = 180^\circ$	$\gamma = 225^\circ$	$\gamma = 270^\circ$	$\gamma = 315^\circ$
90°	$u = 0^\circ$	$u = 45^\circ$	$u = 90^\circ$	$u = 135^\circ$	$u = 180^\circ$	$u = 225^\circ$	$u = 270^\circ$	$u = 315^\circ$
	$\alpha = 90^\circ$	$\alpha = 90^\circ$	$\alpha = 90^\circ$	$\alpha = 90^\circ$	$\alpha = 90^\circ$	$\alpha = 90^\circ$	$\alpha = 90^\circ$	$\alpha = 90^\circ$
	$\gamma = 0^\circ$	$\gamma = 45^\circ$	$\gamma = 90^\circ$	$\gamma = 135^\circ$	$\gamma = 180^\circ$	$\gamma = 225^\circ$	$\gamma = 270^\circ$	$\gamma = 315^\circ$

Table 3.1: Angles of interest for each observation epoch. u is the position in the orbital plane starting at $[0,1,0]$, α is the phase angle and γ is the position from the observer's perspective (see Figure 3.7). The 🌈 symbol denotes $\alpha = 38^\circ$, for which the rainbow feature is visible. This table can be used for reference for plots with flux curves at different observations epochs.

3.3. Rotation

For each of the 8 locations in orbit, each planet is observed for 8 evenly distributed phases of rotation. Rotation of the facets' normal vectors is done with Rodrigues' rotation formula:

$$\mathbf{v}_{\text{rot}} = \mathbf{v} \cos \theta + (\mathbf{k} \times \mathbf{v}) \sin \theta + \mathbf{k}(\mathbf{k} \cdot \mathbf{v})(1 - \cos \theta) \quad (3.8)$$

where \mathbf{k} is the unit vector being rotated about, θ is the angle and \mathbf{v} and \mathbf{v}_{rot} are the normal vector before and after rotation, respectively. First, the North Pole and rotation axis are aligned with θ being the angle between the z-axis and the rotation axis, and \mathbf{k} computed as the cross product of the z-axis and the rotation axis. Next, the normal vectors are rotated about the rotation axis in eight equidistant steps to allow observations of all visible sides of the planet.

As an example, the fluxes of the planet in Figure 3.8 in an edge-on orbit and with a rotation axis of $[0,1,0]$ (90° axial tilt) are shown in Figure 3.9. Since this planet is largely covered by ocean and clouds, the directional and Lambertian curves differ significantly in magnitude, particularly for epochs at the extreme phase angles (38° and 142°). Even though the planet is not homogenous, the values of U are very small and thus would be dominated by noise. For this reason, U is left out from retrievals in subsequent chapters.



Figure 3.8: Planet Earth with a cloud pattern selected from the set created in Section 2.3. The flux curves of this planet can be seen in Figure 3.9.

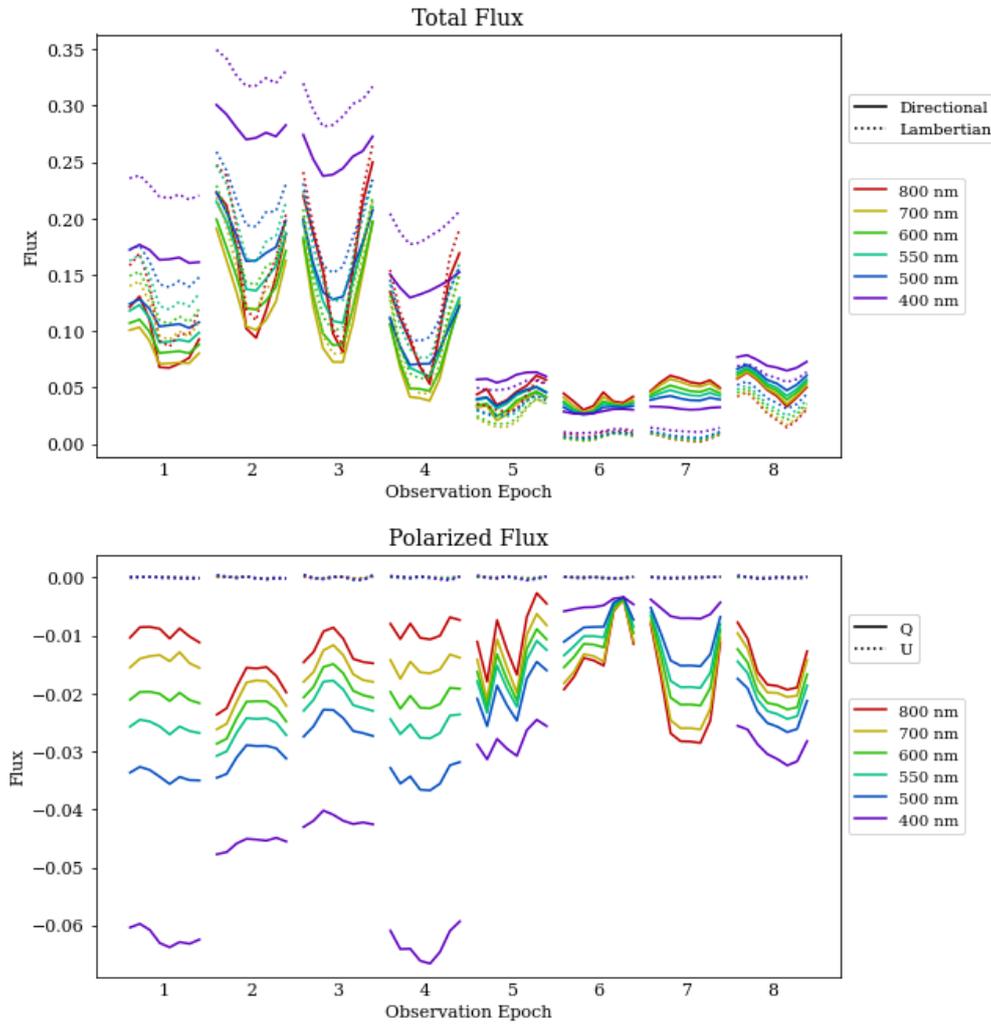


Figure 3.9: Flux curves for the planet shown in Figure 3.8 in an edge-on orbit with axis of $[0,1,0]$ (axial tilt is 90 degrees). See Table 3.1 for the position of the planet during each epoch. The Lambertian assumption leads to an overestimation of the flux at high phase angles (epochs 2 and 3) and an underestimation for low phase angles (epochs 6 and 7). The magnitude of the polarized flux is roughly 5 times lower than the total flux, meaning that it is more susceptible to noise (see Section 3.5). The magnitude of the Stokes parameter U is negligible compared to that of Q and would therefore be dominated by noise and hence it is not used in the retrievals in subsequent chapters.

3.4. Creating a Large Data Set

The neural networks described in Chapter 4 - Chapter 6 require a large data set of planets and their respective light curves for training. In this section, an efficient method to create light curves of each of the 4 million fake planets is described. Instead of calculating each facet's reflection (see Section 3.1) for each planet, orbital location and rotation phase, it is far more efficient to calculate the fluxes for all possibilities shown in Table 3.2 and save these in an 8-dimensional array. The total size of this array is 2.06 billion numbers ($4 \cdot 7 \cdot 64 \cdot 8 \cdot 8 \cdot 1000 \cdot 6 \cdot 3$) and 8.26 Gb when stored as a NumPy float32 array. The fluxes of a planet can then be calculated by summing across all axes (except for inclination), while matching each facet to its surface type.

3.5. Noise

3.5.1. Shot Noise

In this thesis, shot noise is used to simulate noise in the observations. Most exocartography research papers use the unphysical, statistical model of Gaussian noise, since this is a generic form of noise that can be easily compared to e.g. instrumental noise from other papers. However, the noise level is not adjusted to the magnitude of the flux, which can create negative values for observations with a low flux. In comparison, shot noise

Parameter	Number of Possibilities
Surface Types	4
Inclinations	7
Rotation Axes	64
Orbital Locations	8
Phases of Rotation	8
Facets	1000
Wavelengths	6
Stokes Parameters (for directional reflection)	3

Table 3.2: All possibilities that the flux can be calculated be for. By calculating the fluxes of each combination and storing them, the flux of any planet can be computed efficiently by summation. A total of 2.06 billion numbers must be stored.

is the result of the particle nature of light and is the physical lower limit of noise for a perfect instrument with no background noise. Shot noise dictates that the number of photons observed by a detector is drawn from a Poisson distribution with N being the number of photons that would be observed in the absence of this noise, shown in Figure 3.10. This principle is expanded to polarized light by writing I and Q as functions of the fluxes F_{0° and F_{90° , which are linearly polarized in perpendicular directions (i.e. 0° and 90° , respectively), and then rearranging equations:

$$I = F_{0^\circ} + F_{90^\circ}, \quad Q = F_{0^\circ} - F_{90^\circ} \quad (3.9)$$

$$\Rightarrow F_{0^\circ, \text{no noise}} = \frac{I_{\text{no noise}} + Q_{\text{no noise}}}{2}, \quad F_{90^\circ, \text{no noise}} = \frac{I_{\text{no noise}} - Q_{\text{no noise}}}{2} \quad (3.10)$$

Since $I_{\text{no noise}}$ and $Q_{\text{no noise}}$ are known from calculations in the previous sections, one can find $F_{0^\circ, \text{no noise}}$ and $F_{90^\circ, \text{no noise}}$. These are converted from absolute fluxes to numbers of photons by multiplying by N_{max} , the number of photons that would be received from a white Lambertian planet at full phase, given no noise. F_{0° and F_{90° can then be drawn from Poisson distributions and I and Q can be computed with Equation 3.9. The result of adding two Poisson distributions with $N = a$ and $N = b$ is a Poisson distribution with $N = a + b$, so the distribution of I remains unchanged. The result of subtracting two Poisson distribution is a Skellam distribution with $N = a + b$, shown in Figure 3.10 for $a = b$ (and thus centered around 0).

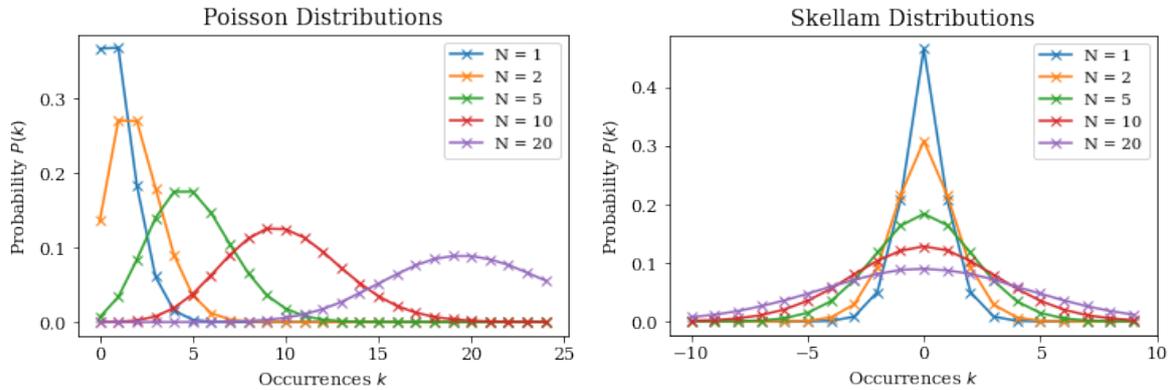


Figure 3.10: The Poisson distribution represents the probability of k total photons being received, given a light source that emits a mean of N photons. The Skellam distribution shows the probability of a polarized flux of k photons being received from an unpolarized light source, since this Skellam distribution is centered around 0.

The effect of shot noise is visualized in Figure 3.11 for two partial sine curves with a maximum magnitude of 0.3 and -0.06 for I and Q , respectively. The magnitudes are roughly representative of those seen in Figure 3.9. Due to the smaller absolute values, the polarization curve is far more sensitive to noise than the total flux curve, with the red line representing 1000 photons in both plots. For this flux curve to be reliably observed only $N_{\text{max}} \approx 500$ photons are needed, while for polarization to be reliably observed $N_{\text{max}} \approx 20,000$ photons are needed.

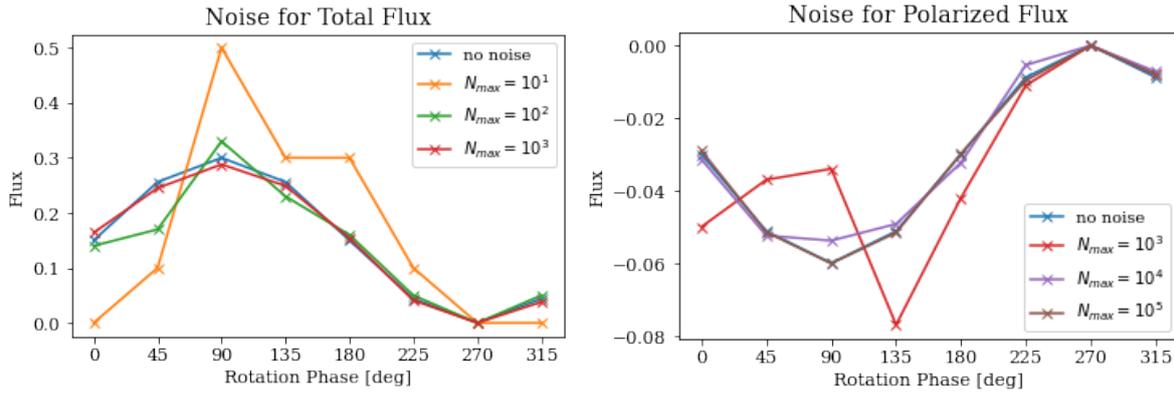


Figure 3.11: Flux curves with noise. The polarized flux is far more sensitive to noise due to its smaller magnitude (see Figure 3.9). N_{\max} is the number of photons corresponding to a flux of 1 (white Lambertian planet at full phase). For illustration purposes, both "no noise" curves are partial sine waves with magnitudes approximating the curves in Figure 3.9.

The signal-to-noise ratio (SNR) of shot noise is dependent on the number of photons that would be received given no noise (Thangjai and Niwitpong [38]):

$$\text{SNR} = \sqrt{N} \quad (3.11)$$

The fluxes of the planets never reach 1, so the SNR (which is unique to each observation) is always less than $\sqrt{N_{\max}}$. For example, a flux of 0.04 (see Figure 3.9) would correspond to $\text{SNR} = 0.2\sqrt{N_{\max}}$.

3.5.2. Shot Noise of Planned Exoplanet Missions

The number of photons that could be received by proposed direct imaging missions is discussed in this section, to assess whether the results in subsequent chapters can be achieved using these mission architectures. The most likely near-future candidates for direct observation of Earth-like exoplanets orbiting Sun-type stars are space telescopes flown in formation with a starshade. This architecture is originally proposed by Cash [4] and makes use of a "starshade" spacecraft that blocks out the star's bright light, allowing a space telescope to directly observe the reflected starlight of its orbiting planets. The starshade must be 30 to 50 meters wide and is flower-shaped with a specific pattern that suppresses diffraction of light around it. In the 2018 report of the NASA Starshade Rendezvous Probe mission by Sara Seager [33], these advantages are listed when comparing a starshade to the advanced on-board coronagraph instrument (CGI) of the Nancy Grace Roman Space Telescope, which will be launched (without a starshade) in 2025:

- Inner working angle reduced from 200-300 milliarcseconds (mas) to ≥ 100 mas
- Flux ratio sensitivity decreased from $5 \cdot 10^{-8} - 5 \cdot 10^{-9}$ (wavelength dependent) to $\geq 10^{10}$
- 10 times larger throughput

For these reasons, a starshade is considered necessary to image Earth-like exoplanets with a space telescope. Next generation ground-based telescopes such as the E-ELT will also have the capability to directly image Earth-like exoplanets around nearby M-type stars but the contrast needed for such planets around Sun-like stars cannot be achieved (Sara Seager [33]). The two starshade missions that have been selected for study are the Starshade Rendezvous Probe that aims to use a starshade in combination with the Nancy Grace Roman Space Telescope before 2030 and the HabEx mission (JPL [19]) that has a dedicated telescope and starshade and aims to launch in the mid 2030s.

To characterize the number of photons that would be received from a nearby exoplanet, the planetary albedo, planetary phase, planetary radius, distance to the observer and effective temperature and radius of the star (we assume black body radiation) all need to be known. To simplify, we assume that the planet is Earth-sized, orbiting at 1 AU distance from a Sun-type star. The planet is assumed to be completely white, Lambertian and at full phase, which corresponds to a flux of 1 (or N_{\max} photons) according to the normalization used for flux computations in Section 3.1.

The integration time of the telescopes is taken to be $24/8 = 3$ hours and the band width of the wavelengths is taken to be 50 nm such that each of the wavelengths that we use in our simulations can be resolved. The

total number of photons emitted by the star for each wavelength band is found by dividing the energy output by the energy of an individual photon. The energy output for each wavelength range is computed using the Stefan-Boltzmann law for the radius and effective atmospheric temperature of the Sun, 695,700 km and 5772 K, respectively.¹² The number of photons reflected towards the observer by the white, Lambertian planet at full phase is equal to:

$$N_{\max} = \dot{N} \cdot t_{\text{integ.}} \cdot \frac{1}{4} \left(\frac{r_{\text{telescope}}}{d_{\text{star}}} \right)^2 \cdot \frac{1}{4} \left(\frac{r_{\text{planet}}}{r_{\text{orbit}}} \right)^2 \cdot \frac{8}{3} \quad (3.12)$$

where \dot{N} is the rate of photons being emitted by the star for the desired wavelength range, $t_{\text{integ.}}$ is the integration time, d_{star} is the distance to the star system and $r_{\text{telescope}}$, r_{planet} , r_{orbit} are the radii of the telescope primary mirror, planet and planetary orbit, respectively. $8/3$ is a correction factor to correct from a homogeneously re-emitting sphere to a Lambertian sphere at full phase (Stam et al. [37]).

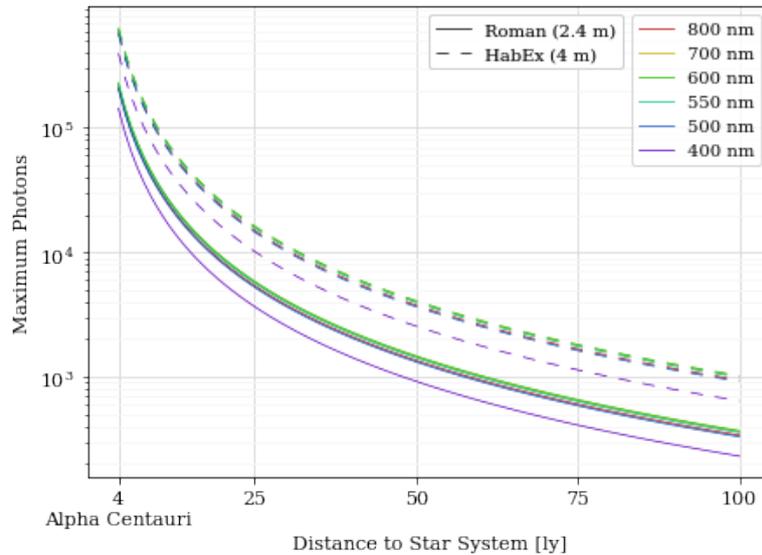


Figure 3.12: The maximum number of photons (white Lambertian planet at full phase) for an Earth-sized planet around a Sun-type star. The integration time is $24/8 = 3$ hours and the wavelength bands are 50 nm wide such that each wavelength can be resolved. The Nancy Grace Roman Space Telescope is shown since a mission has been proposed to fly a starshade in formation with it (Sara Seager [33]). HabEx is a proposed mission for the 2030s with a starshade and a dedicated telescope for direct imaging of exoplanets (JPL [19]). The lines for some wavelengths are covered. The least number of photons is received for a wavelength of 400 nm since the power output of the Sun decreases for this wavelength, while the energy per photon increases. Note that this figure is the same as Figure 1.2 in the introduction.

The results in Figure 3.12 show that the maximum number of photons received from an Earth-like planet would be between 10^5 and 10^6 for a Sun-type star at Alpha Centauri's distance (3 hour integration time). In subsequent chapters, roughly $N_{\max} = 10^4$ photons are needed for accurate retrievals. This would be possible up to a distance of around 20 light years with the Nancy Grace Roman telescope in combination with a starshade and 30 light years for the HabEx telescope, using this simple noise model.

Some of the noise sources that are not included are:

- Background starlight that refracts around the imperfect starshade.
- Background light due to zodiacal and exozodiacal dust. Exozodiacal dust is expected to be common in the habitable zones of other stars and will be the largest source of astrophysical noise (Roberge et al. [31]).
- Instrumental noise.

For these reasons, the noise estimates made in this report should be treated as a lower limit for the true noise. If the SNR of a future mission is too low, it can be increased by increasing the width of the wavelength bands

¹²<https://nssdc.gsfc.nasa.gov/planetary/factsheet/sunfact.html>

or the integration time which would result in lower resolution of the wavelengths and rotational phases, respectively. Another method to increase the SNR is to observe the planet for more epochs than eight days during the year, which requires more time using advanced telescopes that are in high demand.

3.6. Normalization

The total flux observed from an exoplanet is proportional to its albedo and its radius squared (as well as the incident stellar flux that we assume to be equal for all planets in our reflected flux computations). Since an Earth-like exoplanet's radius cannot be constrained from observations of reflected fluxes, a planet with a radius of r and an albedo of a cannot be distinguished from a planet with a radius of $x \cdot r$ and an albedo of a/x^2 for any number $x > 0$. For this reason, relative fluxes rather than absolute fluxes should be used for retrievals. Thus, unless stated otherwise, the retrieval algorithms in subsequent chapters use light curves that are normalized such that their maximum value is equal to 1 by dividing by the largest flux. Normalization is done after applying noise.

4

Rotation Axis Retrieval

The first step taken to retrieve planet maps based on the reflected fluxes calculated as described in Chapter 3, is to estimate the orientations of the rotation axes of the planets. This has been shown by Kawahara [22] to be possible by observing the frequency modulation of the planetary signal over a complete orbit. As the planet orbits its star, the period of the light curve increases or decreases depending on the rotation direction, allowing for the orientation of the rotation axis to be constrained. The approach taken in this thesis does not use this effect, since our planets are "frozen" in their orbital locations for each observation epoch and therefore the frequencies of the light curves are constant.

Instead, we use a convolutional neural network approach (Section 4.1) with 'periodic convolutions', which are modified 1-D convolutions that take advantage of the periodic nature of the light curves. Two layers of periodic convolutions are followed by a series of 5 dense layers with PReLU activation functions and dropout layers to prevent the neural network from overfitting the training data. The outputs of the neural network are the three Cartesian coordinates of the planets' rotation axes.

In Section 4.2, we empirically prove a degeneracy for two planets that are in a side-on orbit, whose maps are flipped about the equator and whose rotation axes are the same in the out-of-orbital plane direction and negatives of each other in the orbital plane. This is tested for several cases and in all cases the same light curves are computed. The neural network is shown to solve this degeneracy for very low levels of noise, which shows that numerical artefacts in the curves can be used to constrain the rotation axis. However, these numerical artefacts can be removed by applying noise to the curves, as discussed in Section 4.4.

The degeneracy discussed above can be solved by constraining the axis to one half of the search space, with the other solution found in the other half of the search space. The results of the retrievals are shown in Section 4.5. Here, we also assess the benefits of detecting polarization and the errors that are caused by assuming Lambertian reflection of a non-Lambertian planet.

4.1. Architecture

4.1.1. Periodic Convolutions

Convolutional neural networks are originally developed for 2-D image recognition (Lecun et al. [27]) but recently, 1-D convolutions have been used to process time-series signals to achieve state-of-the-art results in fields such as "biomedical data classification and early diagnosis, structural health monitoring, anomaly detection and identification in power electronics and electrical motor fault detection" (Kiranyaz et al. [25]). The architecture described in Section 4.1.3 uses 1-D convolutions with some adjustments to take advantage of the periodic nature of the light curves.

Recall that the rotation phases being used are 0° , 45° , 90° ... 315° . Since the prime meridian of the planet (the line of 0° longitude) is arbitrarily chosen, the relationship between the rotation phases of 315° and 0° should be equivalent to the relationship between e.g. 45° and 90° . To make sure that these combinations are not seen differently by the neural network, the first $N - 1$ rotation phases are appended to the end of each light curve before the $1 \times N$ convolutional kernel slides over the light curve, as seen in Figure 4.1 (this figure also shows the definition of kernel size). Thus, for a kernel size $N = 3$, the first 2 rotation phases are appended (i.e. 0° and 45° in the top diagram of Figure 4.1). An additional advantage is that the dimensions of the light curve are not changed by the convolution, without the need for zero-padding. Zero-padding is a common method

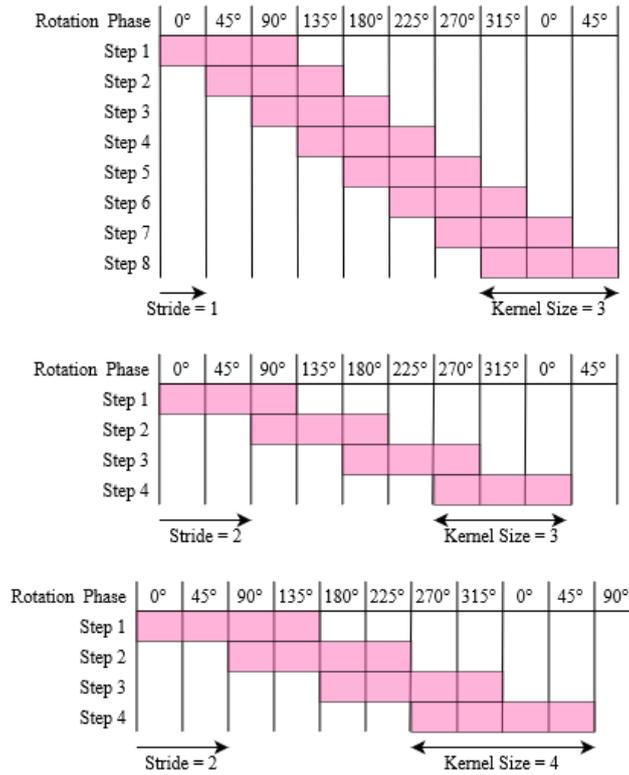


Figure 4.1: Periodic convolutions with different strides and kernel sizes. Since the rotation phases are periodic in nature, the first $N-1$ values are appended to the end of the light curve before the size N kernel slides over, preserving the dimensions (8 steps corresponding to the 8 rotation phases). When a stride of N is used, the kernel size should be a multiple of N so that some rotation phases are not convolved less often than others (as shown for stride = 2 and kernel size = 3). Using this approach reduces the MSE of the neural network by $\approx 10\%$.

of preserving data dimensions by surrounding the data by zeros, which has the disadvantage of increasing susceptibility to spatial bias, as shown by Alsallakh et al. [1].

For down-sampling the dimensions of the data, a stride greater than one can be used (see Figure 4.1 for the definition of stride). However, as shown in Figure 4.1, this can lead to unintended consequences when the kernel size is not a multiple of the stride length, since some rotation phases will then be sampled more often than others, increasing their impact on the output of the neural network. Since for our planets all rotation phases are of equal importance, this is avoided by using a kernel size of 1×4 and a stride of 2 in the final architecture, as illustrated in the bottom diagram of Figure 4.1. We tested the effectiveness of using periodic convolutions by comparing to results of a neural network with normal convolutional layers and the loss (which should be minimized through learning for accurate retrievals) is found to decrease by 10% by using the periodic convolutions.

4.1.2. Inclination Input

The orbital inclination angle of a directly observed exoplanet can be determined by observing the location and speed of the planet with respect to its star. We have tried several architectures that include both the inclination and light curves as inputs to retrieve the rotation axis. Two rough outlines of such approaches are shown in Figure 4.2.

In Figure 4.2a, the light curves are first convolved by 1-D kernels before being concatenated with the inclination angle. To allow the neural network to recognize the relative importance of the inclination in comparison to the light curves, the inclination is multiplied by a factor N that we determine by trial and error. This showed that a factor N of 0 is most effective, leading to the conclusion that this architecture works best without an inclination input.

Another approach is shown in Figure 4.2b. Here we attempted to transform the light curves such that they can be processed by the same convolutions and dense layers for an effective result independent of the inclination. To find this transformation, the inclination is fed into a number of layers that produce an array

that is added to or multiplied by the light curves. For some iterations of the architecture, the light curves are also included as an input for this transformation. It is found that the loss is much greater using this approach than using simple convolutional layers followed by dense layers without inclination as an input.

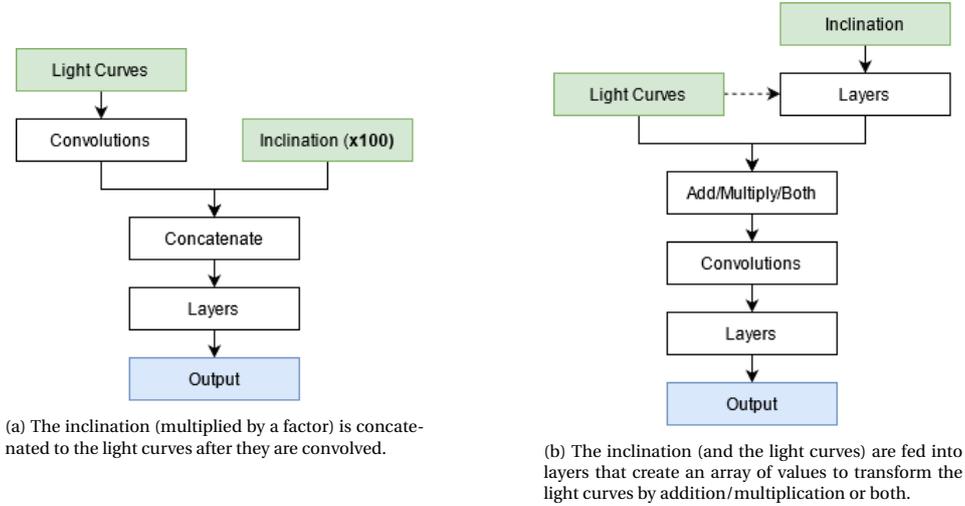


Figure 4.2: Two different approaches to include both inclination and light curves as input for the retrieval of the rotation axis. Both of these attempts proved to be unsuccessful, since removing the inclination input decreased the validation loss of the trained networks.

4.1.3. Final Architecture

Instead of including the inclination as an input, we retrained the neural network for each of the 7 inclinations (0° , 15° , 30° ... 90°). The approach from Figure 4.2a is the most effective one after the inclination input is removed, so this architecture is chosen for the final network. The convolutional layers are called the "feature recognition" part of the network and the dense layers following this are called the "feature combination" part. This architecture is similar to the ones seen in other signal processing neural networks such as in Badshah et al. [3].

The input dimensions to the neural network are either $8 \times 8 \times 6$ without polarization (8 orbital locations, 8 rotation phases, 6 wavelengths) or $8 \times 8 \times 12$ with polarization (both Stokes parameters I and Q are included, which doubles the final dimension). Recall that the Stokes parameter U is of negligible magnitude and is thus not included in the retrievals. The feature recognition part of the network consists of two periodic convolution layers, each with 16 filters. The first layer, which has a kernel size of 3, increases the dimensions of the input to $8 \times 8 \times 16$, as shown in Figure 4.3. The final dimension increases to 16 since each of the 16 filters, which convolve the 6 wavelengths and 3 rotation phases, have an individual output. To halve the size of the data, the second periodic convolution skips half of the "steps" (stride = 2) with a kernel size of 4, leading to a $8 \times 4 \times 16$ output, as shown in the bottom diagram in Figure 4.1. The output of the final convolution layer is then flattened and passed onto the feature combination part of the neural network.

The output from the convolutional layers is fed through 5 dense, fully connected layers with linear activation functions and a bias. These layers have 512, 512, 256, 32 and 3 nodes, respectively. The first four layers are followed by PReLU activation functions to introduce non-linearity (He et al. [15]):

$$f(y_i) = \begin{cases} y_i, & \text{if } y_i > 0 \\ a_i y_i, & \text{if } y_i \leq 0 \end{cases} \quad (4.1)$$

where the slope coefficient a causes non-linearity due to the change in gradient at $y_i = 0$ and is actively learned and manually inspected as described in Section 4.3.2. To prevent overfitting, the first three layers are also followed by dropout regularization layers with a dropout rate of 15%. These dropout layers randomly set 15% of the layer's outputs to 0, which prevents the neural network from overfitting to peculiarities of the training data. The network has a total of 667,907 or 667,619 trainable parameters, depending on whether polarization is included or not, respectively. The trainable parameters are weights of the convolutional kernels, linear nodes or slope coefficients of the PReLU layers, which are all adjusted through learning.

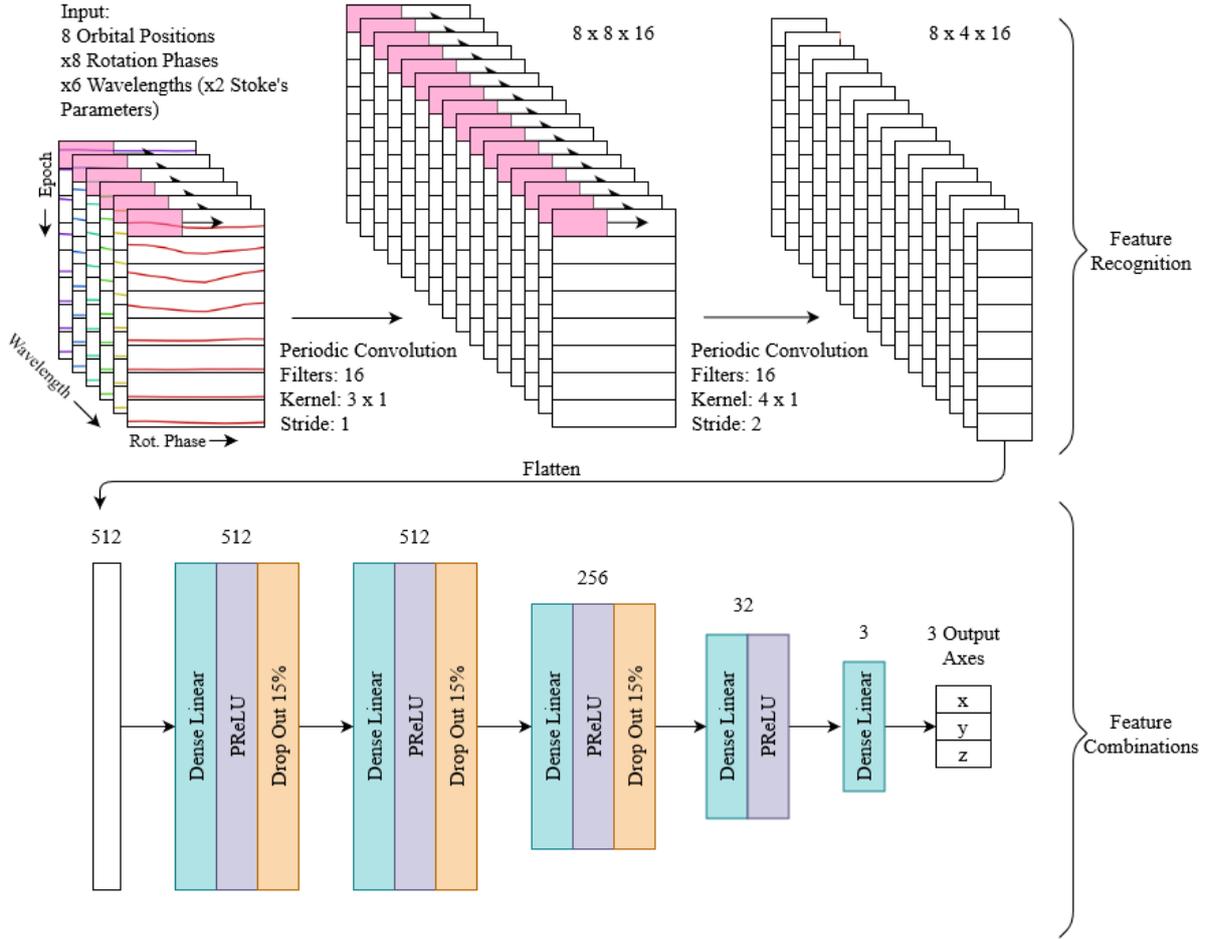


Figure 4.3: The final network architecture to retrieve rotation axes. The feature recognition part uses periodic convolutions and the feature combinations part consists of densely connected layers with PReLU activation functions and dropout layers to prevent overfitting (numbers above lower layers represent number of nodes). In case polarization is used, the shape of the input is $8 \times 8 \times 12$, otherwise it is $8 \times 8 \times 6$. The periodic convolutions maintain the 1st and 2nd dimension of the data since the $N - 1$ first values along the rotation phase axis are appended to the end before convolution. The number of filters determines the 3rd dimension of the output. The total number of trainable parameters is 667,907 or 667,619, depending on whether polarization is included or not.

4.2. Degeneracy

A degeneracy occurs when two planets with different parameters create the same light curves and thus cannot be distinguished from each other. One such degeneracy is identified for planets in side-on orbits, that meet the following two requirements:

- The two planets have the same map except flipped about the equator.
- The rotation axes of both planets have the same out-of-orbital-plane component but the opposite in-orbital-plane component. In the reference system described in Section 3.2 this is expressed as an element-wise multiplication by $[-1, -1, 1]$.

The degeneracy is found because the neural network cannot effectively constrain the x and y axes of planets in a side-on orbit. We identify the above requirements by trial-and-error and confirmed by testing that the light curves are the same for several different pairs of planets that meet both requirements. One example is shown in Figure 4.4, although slight differences in the fluxes can be identified since the maps are not exactly mirrored about the equator due to the asymmetry of the facet scheme about the equator.

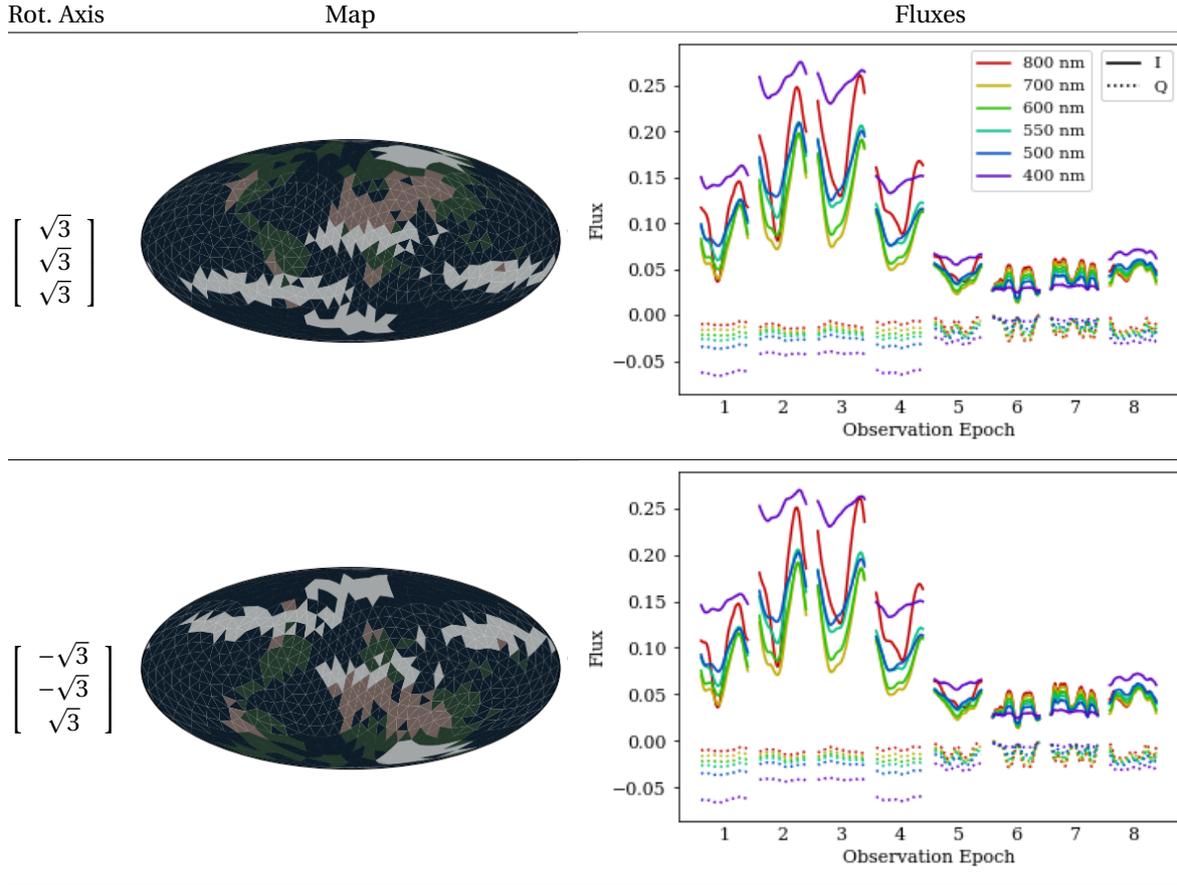


Figure 4.4: Two planets in a side-on orbit with different rotation axes and maps that produce (virtually) the same light curves. This degeneracy occurs when the component of the rotation axis in the orbital plane (the xy plane in this case) is multiplied by -1 and the map of the planet is reflected about the equator. Since the facet scheme on the surface of the planet is not symmetric about the equator, the map is not exactly reflected causing slight differences between the curves, for example at the end of epoch 3 for 400 nm.

4.3. Training

4.3.1. Training Parameters

When the neural network is trained on all planets with a certain orbital inclination angle, the available number of light curves is $4,000,000 / 7 \approx 570,000$. In order to resolve the degeneracy described in Section 4.2, the neural network can also only be trained on all planets with a rotation axis with $y \geq 0$, in which case the number of curves is half of this. Note that we use 90% of the curves for training and 10% are used for validation, with training being stopped when the validation loss does not decrease for 2 consecutive epochs, to prevent overfitting.

Since retrieving the rotation axis is a regression problem, we chose the mean squared error (MSE) as the loss to be minimized by the neural network. We found that for small batch sizes (16 to 64 planets fed into the neural network together for training), the training loss may begin to increase after roughly 10 epochs. We believe that this means the optimization algorithm, Adam (Kingma and Ba [24]), is overshooting the local minimum and adjusting to peculiarities in the small batch, rather than finding the global optimum. This problem was solved by increasing the batch size to 256. The neural network is created and trained using the Keras Python package¹³ and is retrained for each orbital inclination and noise level.

4.3.2. PReLU Slope Coefficients

The slope coefficients a (see Equation 4.1) are inspected after training in Figure 4.5. The second, third and fourth PReLU layers' parameters are distributed about zero, approximating a classical ReLU function, for which $a = 0$. We hypothesized that since these approximate a ReLU function, replacing these PReLU layers

¹³<https://keras.io/>

with ReLU would lead to better results but this was found to be untrue as the validation losses of thus modified networks are higher, meaning that the non-zero a values in these PReLU layers *do* add value. Unlike the other PReLU layers, the first one is not centered around zero and instead has a mean a of -0.32, with a standard deviation that is larger than for the other layers by roughly one order of magnitude. The negative coefficients of the majority of the nodes in this layer mean that negative values returned by the convolutional layers are turned positive when passed in to the feature combinations part of the neural network. It is unclear to us why this is the case.

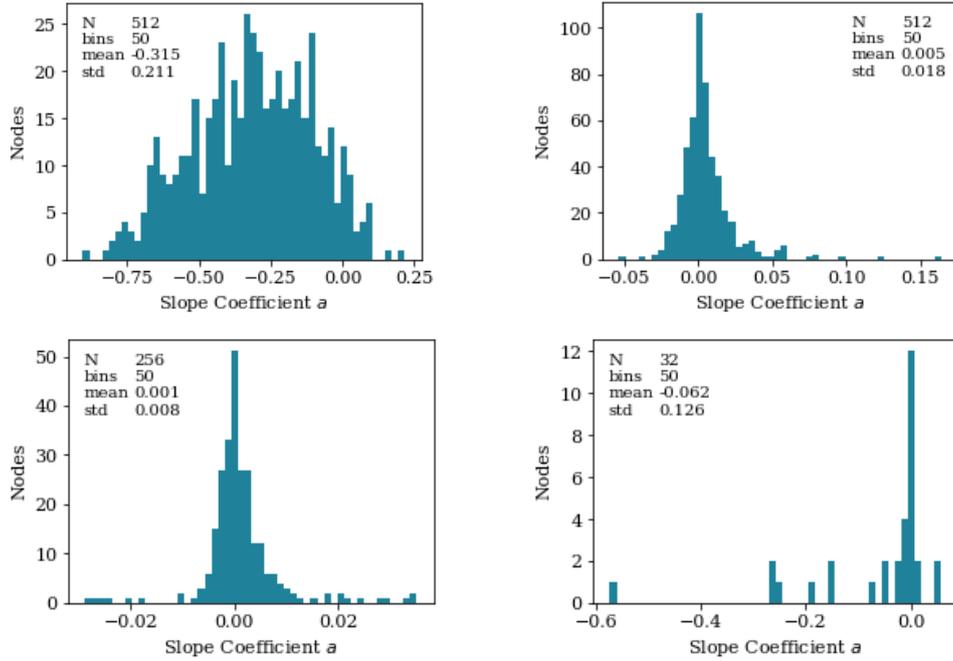


Figure 4.5: PReLU slope coefficients after training (1st, 2nd, 3rd and 4th PReLU layers are top left, top right, bottom left and bottom right, respectively). See Figure 4.3 for the position of each layer in the neural network. These parameters are retrieved from a neural network trained on planets in a side-on orbit with directional light curves including polarization. Although PReLU layers 2, 3 and 4 approximate ReLU layers, replacing them with ReLU leads to worse results.

4.4. Numerical Artefacts

When training the network with light curves with very little noise, the neural network is able to distinguish between the two degenerate cases and correctly estimate x and y of the planet's rotation axis. Since this information is not actually present in the light curves, this shows that the neural network is using numerical artefacts to identify the rotation axis. For example, these numerical artefacts could be due to the way each facet's flux is stored (as a float32) or the facet scheme of the planetary surface.

By increasing the noise levels, these small numerical artefacts can be reduced such that the neural network must use true physical phenomena for the retrieval. By testing that for a given noise level the degenerate cases can indeed not be distinguished, we have confirmed that the neural network solves the retrieval in the intended manner. The results in Figure 4.6 show that a noise level corresponding to $N_{\max} \approx 10,000$ photons is needed.

4.5. Retrieval Accuracy

4.5.1. With Degeneracy

The retrieval accuracy of the neural network is plotted as a function of the planet's orbital inclination angle in Figure 4.7. The loss is highest for side-on orbits, which is explained by the degeneracy since the neural network cannot accurately determine the x and y components of the rotation axes. The loss continually decreases with increasing inclination, likely due to the near-degeneracies that exist for inclinations close to side-on. The differences in between the Lambertian and directional models are similar in Section 4.5.2, and are discussed there.

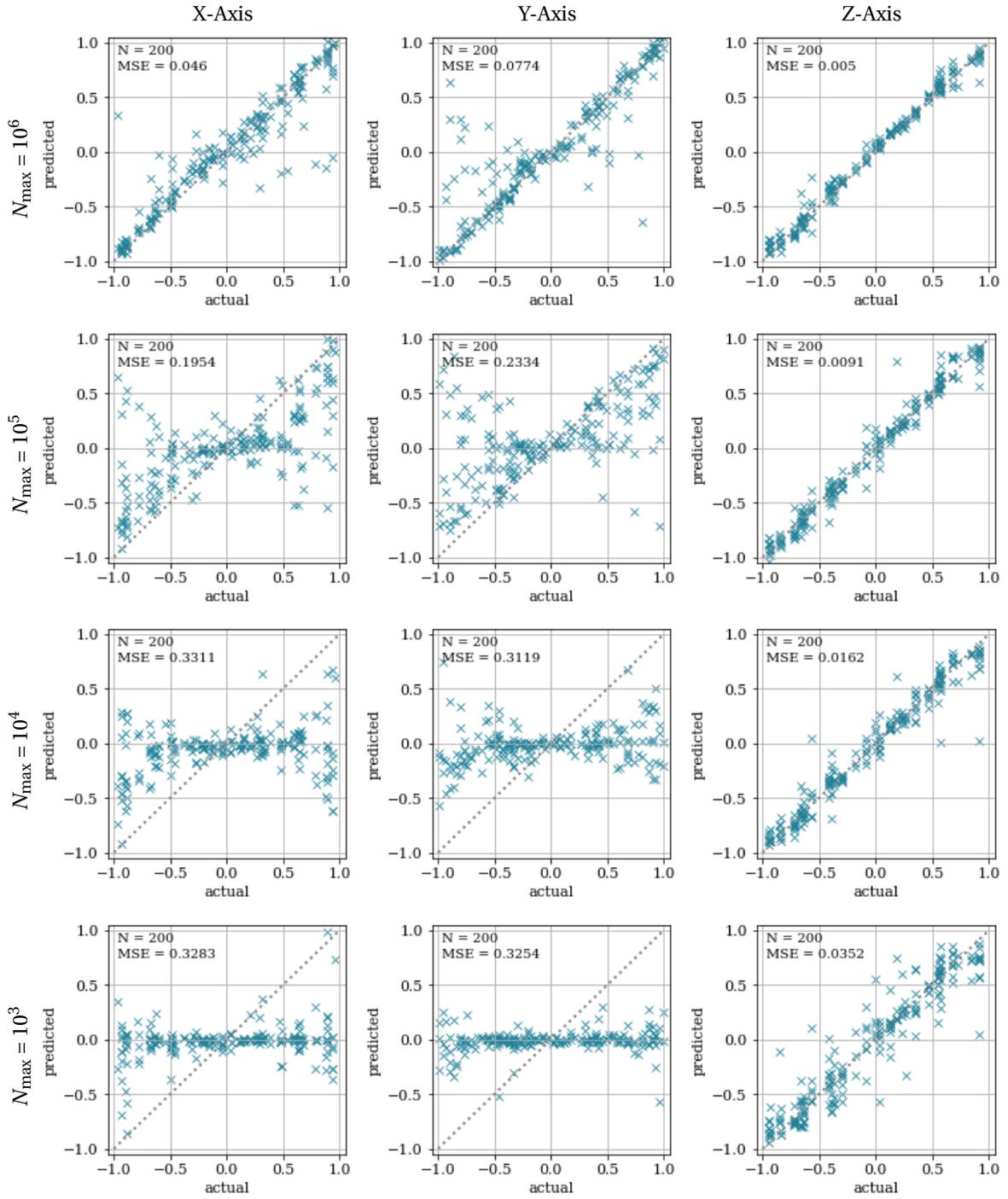


Figure 4.6: Axis predictions for 200 planets in side-on orbits with different noise levels. These plots show that when there is very little noise the neural network is able to resolve the degeneracy described in Section 4.2 using numerical artefacts in the simulated fluxes. For higher levels of noise ($N_{\max} = 10,000$ or 1,000 photons) the degeneracy cannot be resolved so the x and y predictions are grouped around $x = 0$ and $y = 0$, respectively, which minimizes the mean squared error (MSE). This shows that the network uses true physical phenomena for retrieval in these cases.

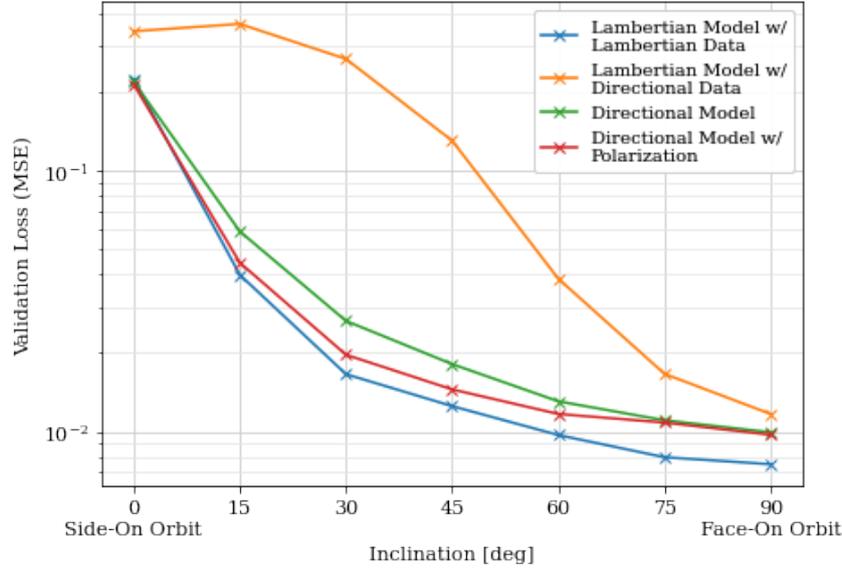


Figure 4.7: The retrieval accuracy as a function of inclination. Since there are no constraints on the rotation axis, the loss is very high for side-on and near-side-on orbits due to the degeneracy and near degeneracies.

4.5.2. Without Degeneracy

The degeneracy can be mitigated by constraining the rotation axes of the planets to $y \geq 0$ and training the neural network on only these planets. The neural network is able to approximate the rotation axis in this half of the search space effectively, as shown by the 100 examples in Figure 4.8. The solution within this search space can then be used to find the solution in the other half of the search space by multiplying the rotation axis by $[-1, -1, 1]$ and flipping the map about the equator. This decreases the MSE of the side-on orbits by roughly one order of magnitude and the MSE of face-on planets by a factor of around 2 (see Figure 4.9). The retrieval is still significantly more accurate for face-on orbits than side-on orbits, with a MSE that is roughly 2 times smaller.

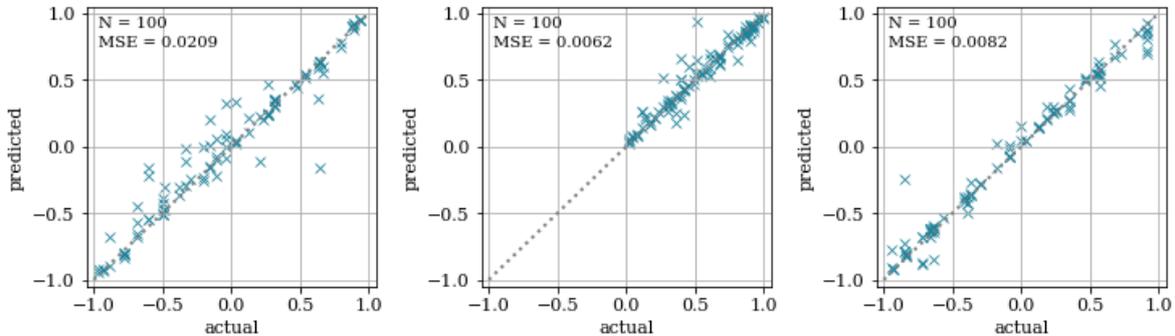


Figure 4.8: Axis predictions (x on the left, y in the center and z on the right) for 100 planets in side-on orbits ($i = 0^\circ$), with the side-on degeneracy mitigated by constraining $y \geq 0$. Once the solution in this search space is identified, the other solution can be found by element-wise multiplying the axis by $[-1, -1, 1]$ and flipping the map about the equator. These figures compare to row 3 in Figure 4.6 where the degeneracy is not visible.

Three models are trained separately: One is trained only on light curves computed using Lambertian reflection and thus "learns" the Lambertian assumption. The other two are trained on light curves computed using directional reflection, once with and once without polarization values included. Note that all losses discussed in this section are validation losses, and are thus the loss of the model when applied to the 10% of validation planets, which it is not trained with. The model trained with Lambertian curves performs best of all combinations when retrieving the rotation axes of Lambertian planets, with an MSE of 0.0049 for $i = 75^\circ$. However, this is shown to be an inaccurate retrieval since when the model is applied to the more accurate directional light curves, the loss is drastically higher, by roughly one order of magnitude for low inclinations.

This shows that if the neural network "learns" the Lambertian assumption, this leads to large errors for non-Lambertian planets at low inclinations. Evaluating such a model using only light curves computed using the Lambertian assumption would lead to incorrect confidence in the retrieval algorithm's accuracy. However, we show that when such a retrieval algorithm is used to retrieve the rotation axes of non-Lambertian planets in a face-on orbit, the model has an MSE of 0.0085, only 32% greater than 0.0064 for the model trained trained on light curves computed using directional reflection. This confirms that the reflected flux in a side-on orbit is mostly Lambertian.

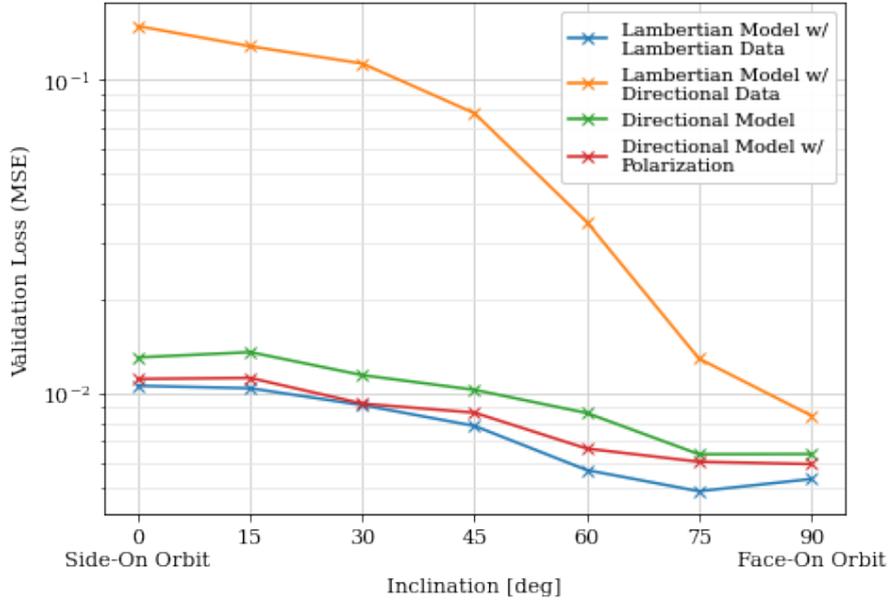


Figure 4.9: The retrieval accuracy as a function of inclination after resolving the degeneracy by constraining $y \geq 0$. The loss for low inclination is significantly lower than in Figure 4.7, where the degeneracy is not resolved. The graph shows that the Lambertian assumption causes very large errors, as a neural network trained on Lambertian curves and applied to more accurate directional light curves has an MSE roughly one order of magnitude larger than a model trained on directional light curves. For a side-on orbit the effect is relatively small (MSE only 32% larger). Including polarization increases the accuracy of the retrievals for all inclinations.

The MSE of the directional models is also highest for the side-on orbit and decreases (nearly) continuously until reaching a minimum for the face-on inclination, by a factor 2. This shows that planets in side-on or near side-on orbits are more difficult to characterize in terms of their rotation axes' orientations. We show that including polarization measurements in the retrievals increases the accuracy of the retrieval for all orbital inclinations. The beneficial effect of including polarization in the retrieval is smallest for the inclinations of 75° and 90° , which can be explained since the reflected flux for these inclinations is mostly Lambertian. The largest decrease in MSE by including polarization occurs for $i = 60^\circ$, where including polarization decreases the loss by 23%.

4.6. Verification

To verify that the retrieval algorithm is able to retrieve planets with maps and rotation axes not included in the training data, light curves are calculated for a model planet Earth with the cloud pattern in Figure 3.8. The fluxes are computed using directional reflection with polarization for a face-on orbit with a rotation axis of $[0.918, 0.281, -0.281]$, such that the tilt is 23.4° . Noise corresponding to $N_{\max} = 10,000$ photons is then added to the curves and they are normalized to have a maximum value of 1, so that the relative fluxes are used (since the radius of an exoplanet cannot be constrained using observations of reflected starlight). The neural network trained on planets with rotation axes with $y \geq 0$ predicts a rotation axis of $[0.904, 0.170, -0.331]$ for these light curves, corresponding to a MSE of 0.0050, similar to the MSE shown for face-on orbits in Figure 4.9. Thus, we can conclude that the neural network has not "memorized" the sets of maps or rotation axes in the training data and constrains the rotation axis in the intended manner.

5

Albedo Map Retrieval

In this chapter, we show our retrievals of albedo maps from light curves like the ones retrieved by other authors (see for example Fujii and Kawahara [8], Kawahara and Masuda [23], Fan et al. [6], Farr et al. [7], Asensio Ramos and Pallé [2]) and in Section 5.1 we show that this is possible using simple neural networks with one to three layers. The mean squared errors (MSE) of the retrieved maps vary between 0.015 and 0.022, depending on the combination of rotation axis and orbital plane (Figure 5.2). For side-on orbits, rotation axes with a tilt near 90° are optimal for retrieval of albedo maps and for face-on orbits, rotation axes with a tilt near 0° (normal to the orbital plane) are optimal.

In Section 5.3, the architecture is expanded to relative (normalized) light curves. Three layers with ReLU activation functions are used to estimate two scaling factors, one that is multiplied by the retrieved map and one that is added to the retrieved map to correct for the overall brightness of the planet. The MSE of the retrieved maps increases to only 0.016 for the ideal geometry when using this architecture, compared to an MSE of 0.015. Applying noise to the curves decreases the retrieval accuracy (Figure 5.4), but some features of Earth, like North America, are still visible for maximum photon numbers greater than or equal to 10,000.

Finally, we test the Lambertian assumption by applying a neural network trained with light curves computed using Lambertian reflection to light curves computed using directional reflection. The neural network "learns" the Lambertian assumption and retrieves non-existent concentric patterns about the poles of the non-Lambertian planets. These artefacts are shown in Figure 6.4 and are retrieved for all orbital inclinations except for face-on orbits.

5.1. Absolute Light Curves

In this section, we attempted to replicate the results of other authors to retrieve the albedo map of a planet based on reflected light curves that are computed using Lambertian reflection and are not normalized. Since this is a linear problem, an estimation of the solution can be found with very simple neural networks.

Over the course of one orbit around the star, the complete surface of a planet is visible to an observer only in the special case that the rotation axis is normal to the vector to the observer. Since information about facets that never face the observer is not present in the light curves, the neural network cannot retrieve the albedos of those facets, and they are not included as outputs. The facets that will be visible to the observer can be easily identified by checking whether the facet's normal vector has a positive x component for any combination of orbit location and rotation phase. The sum of facets that fulfill this requirement is equal to the number of facet albedos retrieved by the neural network. This number can be between 500 (the rotation axis is pointing directly towards or away from the observer and half of the facets become visible) and 1000 (the rotation axis is normal to the vector towards the observer and each facet becomes visible for some rotation phase depending on the orbital location). Note that every facet that becomes visible also becomes illuminated while being visible.

Since the problem is linear in nature, the first architecture that is tried is a single layer of nodes (see Table 5.1). Each facet output is thus computed by multiplying the 64 flux values (8 orbit locations \times 8 rotation phases) by 64 trainable parameters. It is found that including biases in the nodes decreases the accuracy since there is no bias of any facet towards a higher or lower albedo, so these are not included. The number of trainable parameters for this single-layer architecture is 64 multiplied by the number of outputs (the number

of visible facets).

To test this architecture and compare it to others, it is trained using Lambertian, non-normalized flux curves for an orbital inclination of 30° and a rotation axis of $[0.93, -0.20, -0.30]$. This geometry is found to be the best combination in Section 5.2. The wavelength chosen for the retrievals is 550 nm, since at this wavelength it is easiest to distinguish the albedos of all 4 surface types from each other (see Figure 3.4). Furthermore, 10% of the data is used as validation data, the batch size of the training is set to 32 (found by trial and error to be the best), the network is trained until the loss does not decrease for 10 consecutive epochs and the loss function used is the mean squared error (MSE) since this is a regression problem. The training parameters are not changed for the other architectures discussed in this section so that comparisons are fair.

Model Architecture	Val. Loss (MSE)	Training Epochs
681 Nodes	0.0186	413
64 Nodes	0.0164	249
681 Nodes		
Periodic Convolution		
64 Nodes	0.0146	383
681 Nodes		

Table 5.1: Three of the tested architectures to retrieve facet albedos. These architectures are tested for the best combination of inclination and rotation axis (shown in Figure 5.2), for which 681 facets are visible. The inputs to the neural network are 64 Lambertian fluxes (8 orbit locations \times 8 rotation phases) for the wavelength of 550 nm. The architectures are trained until the validation loss does not decrease for 10 consecutive epochs. The periodic convolution uses a 1×3 kernel, as shown in Figure 4.1.

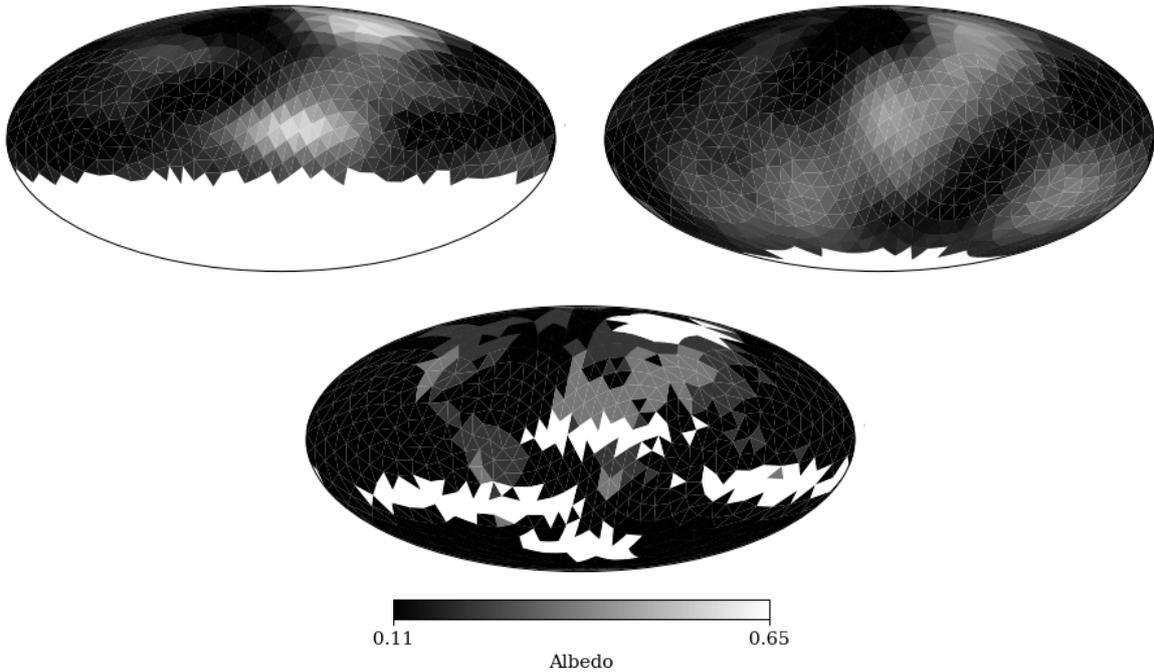


Figure 5.1: Examples of albedo map retrievals for the best (top left) and worst (top right) geometries described in Figure 5.2 using absolute light curves for a wavelength of 550 nm. 681 and 917 facets are visible in the two cases, respectively. The map of the planet being retrieved are the model Earth at a wavelength of 550 nm (bottom). The mean square error for these two retrievals is 0.0286 and 0.0129, respectively, which is similar to the MSE of the validation data for these geometries (see Figure 5.2).

The validation loss of the single layer after 413 training epochs is 0.0186 (see Figure 5.4 for a visual representation of what retrieved maps look like for different loss levels). By trial and error, it is found that adding another layer of 64 nodes (equal to the number of flux inputs) decreases the MSE to 0.0164 while also roughly halving the number of training epochs to 249. Adding more dense layers than this does not further decrease the validation loss of the network. However, the accuracy can be improved by including a periodic convolution (described in Section 4.1.1) with a kernel size of 1×3 before the two densely connected layers. The MSE

of this model when applied to the validation data is equal to 0.0146. Two example retrievals are shown in Figure 5.1 for a model planet Earth in the best and worst configurations of inclination and rotation axis, which are described in Figure 5.2.

5.2. Retrieval Accuracies for Different Inclinations and Rotation Axes

In this section, the effect of inclination and rotation axis on the retrieval accuracy of the albedo maps is investigated. Only half of the 64 axes in the training data are discussed since all rotation axes with $x \leq 0$ can be reflected in the zy plane to create a new axis with $x \geq 0$ that has the exact same observations but in reverse order and hence the same retrieval accuracy would be found.

The final model from Table 5.1 is re-trained for all 32 axes with $x \geq 0$ for each of the 7 inclinations (0° , 15° , 30° , 45° , 60° , 75° and 90°). The MSE of the validation data is plotted for each combination in Figure 5.2 with the locations of the crosses indicating the orientations of the rotation axes for each orbital inclination angle i . Note that the number of facets that are retrieved varies for each rotation axis, since not all facets become visible.

We find that a rotation axis angle (angle between the rotation axis and orbital normal vector) near 90° provides the best retrieval accuracy for side-on and near-side-on orbits (i.e. $i = 0^\circ$ and $i = 15^\circ$). For side-on orbits (in the xy plane), the orbital movement modulates the signal across the y (vertical) axis of the planet's surface. When the rotation axis has a large component in the orbital plane, this provides modulation in the opposite (z) direction. Conversely, when the axis is normal to the orbital plane the modulation due to the rotation of the planet is in the same direction as the modulation due to the orbital movement. Side-on orbits with rotation axes normal to orbital plane thus have the worst accuracy of all combinations. The worst geometry that is studied is for a side-on orbit and a rotation axis of $[0.34, -0.11, -0.94]$, with an MSE of 0.022.

For face-on and near-face-on orbits, the orbital movement of the planet modulates the signal in both the y and z directions. The best retrieval accuracies are then found for axes near to the normal of the orbital plane

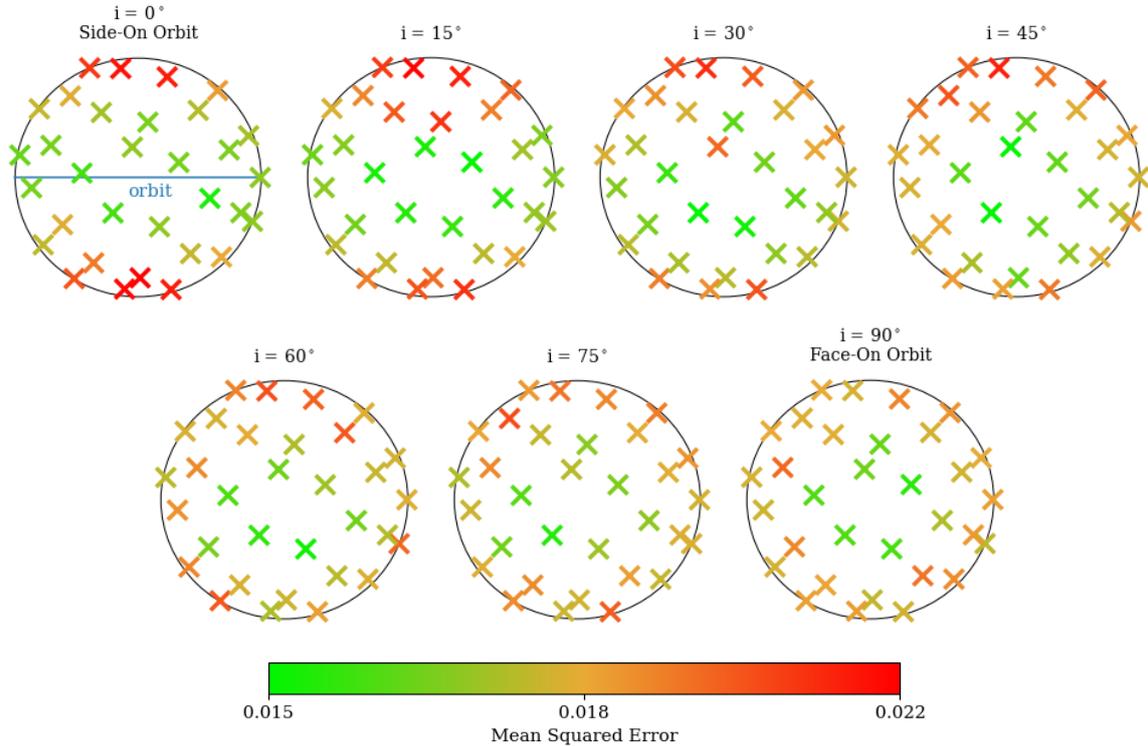


Figure 5.2: Accuracy of albedo retrievals using the final architecture from Table 5.1 for each combination of rotation axis and inclination. The axes are projected onto the yz plane (y and z are the horizontal and vertical axes, respectively). Side-on orbits are in the xy plane and face-on orbits are in the zy plane. Only axes with $x > 0$ are shown since the others are mirrors with the rotation epochs in reverse order. The accuracies are generally higher for rotation axes that have a large tilt for (near-)side-on orbits and a small tilt for (near-)face-on orbits. The highest loss is found for $i = 0^\circ$ and an axis of $[0.34, -0.11, -0.94]$, while the lowest loss is found for $i = 30^\circ$ and an axis of $[0.93, -0.20, -0.30]$.

that also modulate in both directions. Axial tilts near to 90° modulate in only one direction and thus show slightly worse results. The best geometry is found for an inclination of 30° and an axis of $[0.93, -0.20, -0.30]$, as in this case the orbit and rotation axis both modulate across two perpendicular directions. The rotation axis angle is 78° , so the rotation axis lies close to the orbital plane, roughly in the direction of the observer. The MSE for this case is 0.015 and this geometry is chosen for many of the remaining example retrievals in this thesis.

5.3. Relative (Normalized) Light Curves

As is discussed in Section 3.6, flux curves that are normalized to an arbitrary maximum value (in our case 1) should be used when developing retrieval algorithms for directly observed exoplanets since the radius of an exoplanet is difficult to constrain from direct detections. To do this, we propose a new, modified architecture in Figure 5.3. The outputs of the periodic convolution are used to estimate the overall brightness of the albedo maps by three densely connected layers with 32 nodes each. Two single nodes, which output a single number each, are then multiplied and added to the albedo map to scale the map to the estimated brightness. The densely connected layers have ReLU activation functions, which are a special case of PReLU (Equation 4.1) where $a = 0$. Based on our experience, using PReLU layers where a is actively trained instead of ReLU layers in this architecture does not provide better results and increases the number of epochs needed to reach the same validation loss.

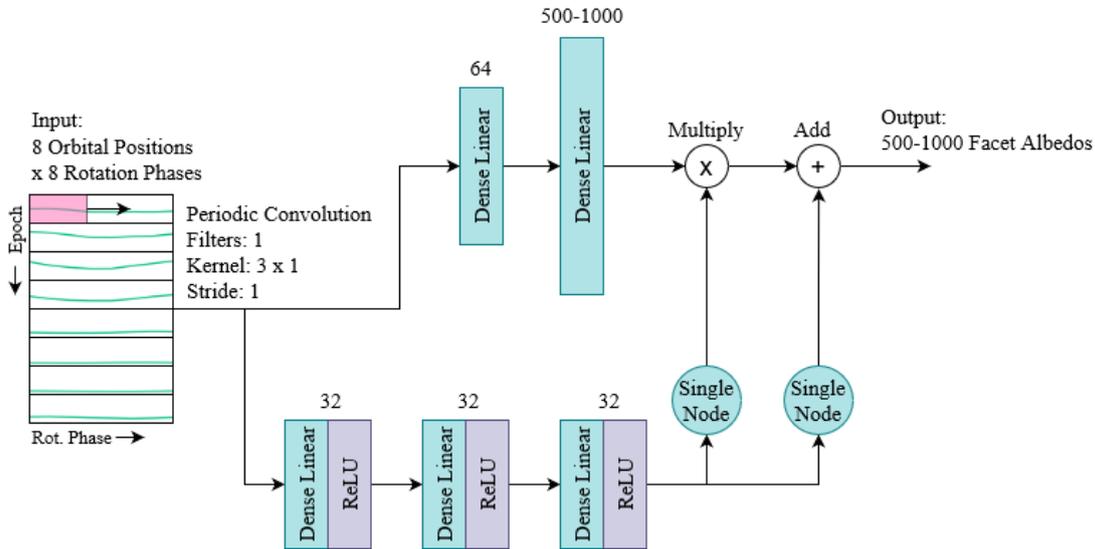


Figure 5.3: The architecture used to retrieve relative light curves that have been normalized such that their maximum value along the orbit is 1. Several "logic" layers with ReLU activation functions are used to scale the output map by adding and multiplying by two single values. The scaling part of the network decreases the loss from 0.0197 to 0.0162 (for the ideal geometry described in Figure 5.2). The output of the neural network is between 500 and 1000 values, depending on the number of facets that is visible for the specific geometry. Neurons in the bottom layers have biases but ones in the top layers do not.

When using the final neural network that is designed for absolute flux curves (Table 5.1) and training it with normalized flux curves, the MSE increases from 0.0146 to 0.0197. By using the scaling discussed above, the loss function can be decreased from 0.0197 to 0.0162. This shows that the neural network is able to effectively estimate the brightness of the map. Examples of retrievals using this architecture are discussed in the next section.

5.4. Effects of Noise

In this section, the effect of noise on the albedo map retrievals is studied. We train the architecture from Figure 5.3 on normalized, Lambertian light curves with different levels of shot noise (explained in Section 3.5). The MSE of the validation data and the MSE of retrieved Earth maps are both shown Figure 5.4. For all noise levels the retrieval of Earth is significantly better than for the validation data. This could be because the Earth map is largely covered by vegetation and ocean, which have a similar albedo for a wavelength of 550 nm (relative to e.g. a combination of clouds and ocean.) Homogeneous planets may be easier to retrieve since

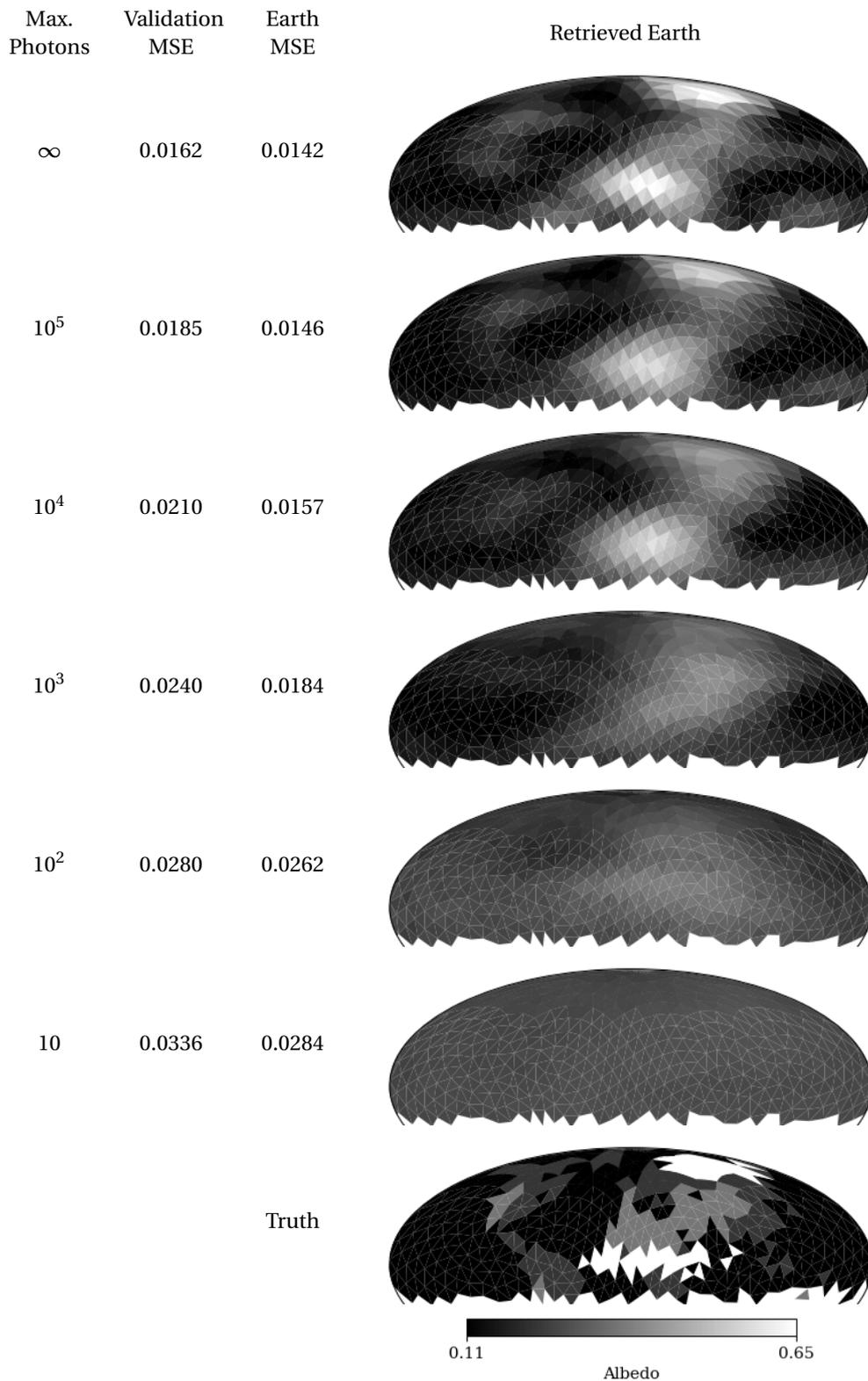


Figure 5.4: Albedo retrievals of Earth for different levels of noise (see Figure 3.11), using the optimal configuration from Figure 5.2. The architecture from Figure 5.3 for normalized flux curves is used and the flux curves are computed using the Lambertian reflection model. The validation MSE refers to the MSE when the model is applied to the 10% of fake planets that are used for validation, rather than training. The mean squared error for the Earth retrieval is lower than for the validation data for all noise levels. At a noise level of $N_{\max} = 100$ photons the clouds are retrieved as high albedo regions and at a noise level of $N_{\max} = 10,000$ photons Afro-Eurasia and North America can both be distinguished.

there are less hard-to-predict details with large penalties for the loss function.

For the case with no noise ($N_{\max} = \infty$), the clouds, the Sahara and North America can be clearly distinguished. These continental features disappear at a noise level of $N_{\max} \leq 1000$ photons. For a noise level of $N_{\max} = 10$ photons, the overall albedo of the planet can be estimated but the map is not spatially resolved. The results shown here are comparable to results by other authors (Fujii and Kawahara [8], Kawahara and Masuda [23], Fan et al. [6], Farr et al. [7], Asensio Ramos and Pallé [2]) although precise comparison is difficult due to differences in geometry, noise models, number of observations and map of the planet to be retrieved.

5.5. Errors due to Lambertian Assumption

Since all other authors, to our knowledge, use Lambertian reflection for their retrievals, it is interesting to evaluate the validity of the Lambertian assumption. To do this, the model from Figure 5.3, which provides comparable results to those by other authors, is trained on Lambertian light curves and then applied to directional light curves (i.e. light curves computed using directional reflection). In Section 4.5, we showed that the Lambertian assumption can only be used for rotation axis retrievals of planets in face-on orbits, so it is expected that the same holds for retrievals of planet maps.

As can be seen in Figure 5.5, when the Lambertian assumption is used to retrieve planets based on directional curves, strong concentric artefacts about the pole appear for all inclinations besides the face-on

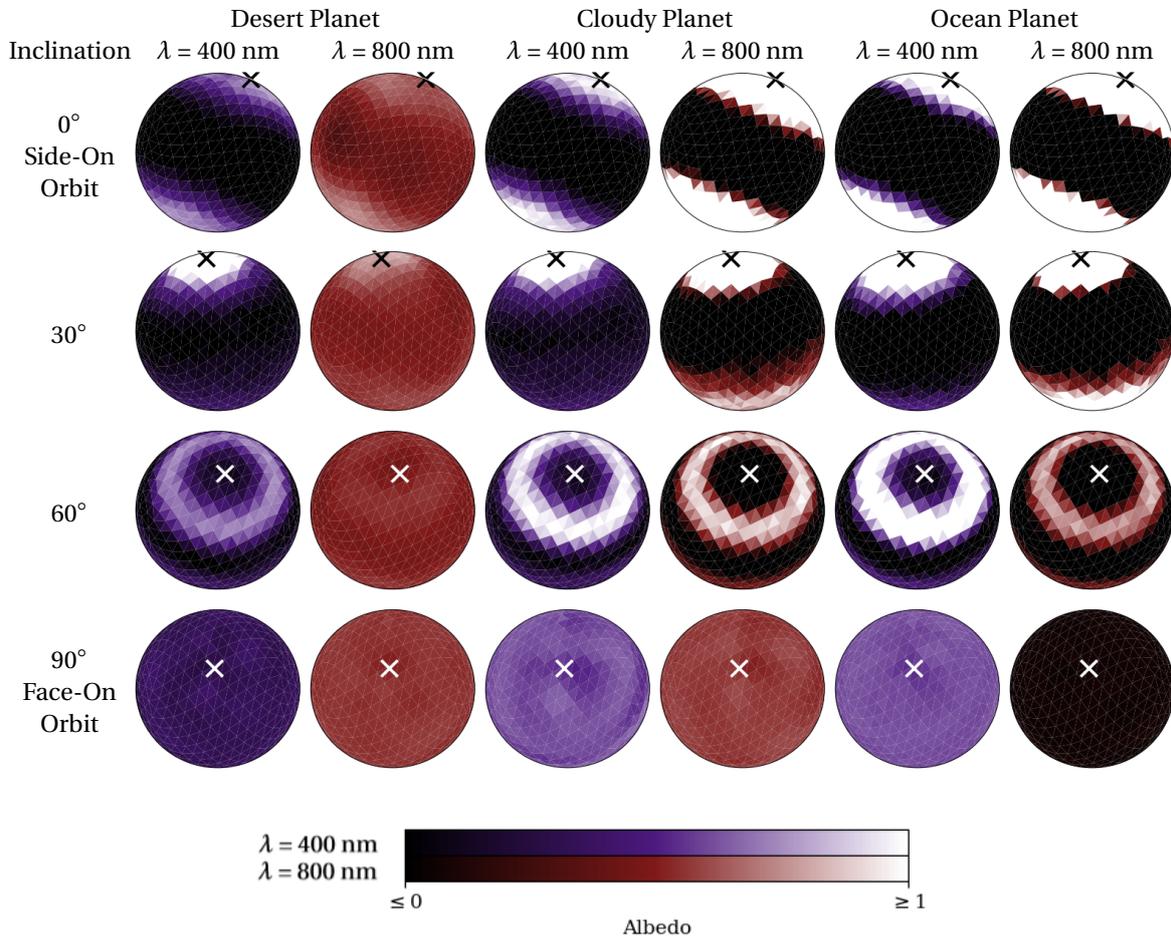


Figure 5.5: Albedo retrievals of directional light curves for homogeneous desert, cloud and ocean planets using the architecture from Figure 5.3 trained on Lambertian light curves. The axis (shown with a white or black X) for each inclination is chosen from Figure 2.13 to minimize the difference to the normal of the orbital plane. The tilts of the axes are 23° , 11° , 5° and 15° for the inclinations of 0° , 30° , 60° and 90° , respectively. Concentric patterns are observed around the axis for all inclinations besides face-on, due to the lower total flux at high phase angles and higher total flux at low phase angles of directional light curves compared to Lambertian light curves. This shows that making the Lambertian assumption leads to incorrect retrievals for planets that are not in a face-on orbit. The retrievals of the desert planet at $\lambda = 800$ nm are best since the desert is modelled as a Lambertian surface and atmospheric effects are minimal at this wavelength.

orbit. Each of the planets in the figure is a homogeneous planet and should be retrieved as such, which is only the case for the face-on orbit. The artefacts at other inclinations are due to the overestimation of the flux at low phase angles and underestimation of the flux at high phase angles when using the Lambertian assumption (see Figure 3.5 for a comparison of Lambertian and directional light curves). Since the planets are homogenous and thus have the same flux at all rotation phases, the magnitude at each phase angle is the only difference between the Lambertian and directional reflection models in this case.

Since most exoplanets are not in a face-on orbit, these errors demonstrate a need for new retrieval algorithms that use directional light curves to map planet surfaces.

6

Surface Type Map Retrieval

Since albedo maps, as used by the other authors working on this topic, do not completely describe a planet's surface and do not distinguish unique non-Lambertian features like oceans and clouds, a new approach is used for creating planet maps. Instead of retrieving an albedo value in the interval $[0,1]$ for each surface facet, as is done in Chapter 5, the *surface type* of each facet is retrieved. The four possibilities for each facet are ocean, clouds, vegetation and sandy desert. For each facet, our neural network predicts a probability for each possibility.

To perform regularization of the surface map across the spherical surface of the planet, we use spherical convolutions, as is done by Asensio Ramos and Pallé [2]. The algorithm by Krachmalnicoff and Tomasi [26] is adapted so that it can be used for the Fibonacci sphere. We improve the algorithm by repeating and taking out some values to take advantage of the periodic circular shape, rather than using zero-padding as is done by Krachmalnicoff and Tomasi [26], which can cause artefacts (Alsallakh et al. [1]). The spherical convolutions are applied after two layers of periodic convolutions and a dense layer, as shown in Figure 6.2.

We find that our proposed architecture predicts 87% of facets correctly for the ideal geometry in the absence of noise. Increasing the noise decreases the retrieval accuracy (see Figure 6.3) but most of Earth's features such as the Americas, the Sahara Desert and Europe can all be distinguished even for $N_{\max} = 100$ photons (see Figure 6.5.) These surface type maps show more detail and are more resilient to noise than the albedo maps discussed in Chapter 5. In Section 6.4 the benefits of polarization are discussed. Due to their unique polarization signature, including the polarization of the reflected starlight in the retrievals increases the retrieval accuracies of ocean and cloudy facets by 2% and 1%, respectively.

As was also the case for the albedo maps (see Chapter 5), using a neural network trained on Lambertian light curves (called the Lambertian neural network) to retrieve non-Lambertian planets, results in highly erroneous maps (see Figure 6.4.) However, in Figure 6.6 it is shown that the classification accuracies of the Lambertian neural network are above 70% for all four surface types when the orbit of the planet is face-on, since for this geometry the reflected light is mostly Lambertian.

6.1. Architecture

6.1.1. Spherical Convolutions

2-D convolutions are originally developed for image recognition (Lecun et al. [27]) but can also be used for 2-D image generation, for example when generating faces (Karras et al. [21]) or for semantic segmentation, where the output of the neural network is an image with each pixel belonging to a class (Wang et al. [41]). Asensio Ramos and Pallé [2] demonstrate that convolutions on the surface of a sphere can be used to regularize retrieved exoplanet maps. They use the spherical convolution algorithm developed by Krachmalnicoff and Tomasi [26] together with ReLU activation functions to regularize a retrieved planet map on a HEALPix pixelization scheme (Gorski et al. [11]).

Since the HEALPix scheme is not used in this thesis, we have adapted the spherical convolution algorithm by Krachmalnicoff and Tomasi [26] to a Fibonacci sphere. The spherical convolution takes advantage of the optimization that has been achieved for 1-D convolutions by expanding the 1-D list of facets on the sphere's surface such that each facet is followed by its N surrounding facets. Then a kernel with size and stride equal to $N + 1$ (kernel size and stride are illustrated in Figure 4.1) convolve each facet with its surrounding facets.

To achieve this for a Fibonacci sphere, each facet's surrounding facets are identified by checking which other facets share a vertex. They are then ordered in clockwise direction by calculating the clockwise angle from the z-direction to each surrounding facet's center (after projecting into the plane of the center facet.) This series of facets constitutes a kernel with 1 "ring." More rings can be added by taking the surrounding facets of the inner ring. A problem that needs to be solved is that the rings around the facets do not all have the same size but the 1-D kernel can only have one size. To solve this, Krachmalnicoff and Tomasi [26] add zero values to the end of the series that represents one ring. This method, called zero-padding, has been shown by Alsallakh et al. [1] to potentially cause unwanted artefacts for regular 2-D convolutions. Hence, we improve the solution by adding the first value to the end of the series, since the rings are periodic in nature.

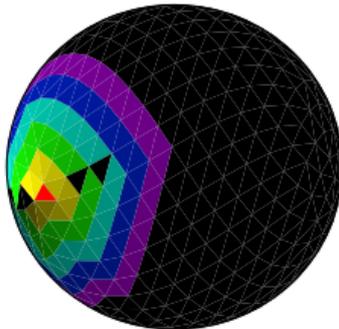


Figure 6.1: An example of a spherical convolution with 5 rings. The method developed by Krachmalnicoff and Tomasi [26] (originally for HEALPix) is adapted to the Fibonacci sphere with the improvement that the periodic nature of rings is used rather than zero-padding. Some facets inside the rings are black because they are not included in the kernel, since the number of facets in each respective ring must match for all kernel locations on the sphere.

The mean integer value of the rings surrounding each facet are computed and for each facet's rings, if the number of facets in the ring is not equal to the mean, facets are either repeated or taken out in an evenly spaced manner. This leads to some "holes" in the rings as shown in Figure 6.1. In other places there are double counts. These holes and double counts could lead to unintended artefacts.

6.1.2. Architecture

To create surface type maps, the output dimensions of the neural network should equal the number of visible facets (500-1000) times the number of possible surface types (4). When creating a visual map or checking the classification accuracy, each facet's predicted surface type is the one with the highest probability.

To recognize patterns in the light curves, a similar periodic convolution approach as in Figure 4.3 is used. Two periodic convolutions (see Section 4.1.1) with 16 kernels of size 1x3 and 1x4 are applied to the light curves. No down-sampling is done as the resolution along the rotation phase should stay high to create accurate maps (stride = 1). This means that we used a vector of size 1024 after flattening the outputs of the periodic convolutions. This is followed by one dense layer for each surface type. Each dense layer has as many nodes as there are output facets.

Finally, spherical convolutions are used for regularization of the surface map. A ResNet approach (He et al. [16]) is used so that the dense layer output is fed into spherical convolutions (four filters, one ring) and the output of the spherical convolution is then added to the original output. This is done 5 times before the final output of the neural network. The spherical convolution kernels are set to all zeros before training, such that the spherical convolutions minimally interfere with the learning of the earlier layers.

6.1.3. Training

The neural network is trained for one combination of inclination and rotation axis at a time, so there are roughly $4,000,000 / 7 / 64 \approx 9000$ light curves in the training data, of which 10% are used for validation. The neural network is trained using the Adam optimization algorithm (Kingma and Ba [24]), until the validation loss does not decrease for 5 consecutive epochs. We find that a batch size (number of curves fed into the neural network simultaneously for training) of 32 works best, since larger batch sizes can cause the neural network to overshoot the local minima. We hypothesize that this is due to the gradient magnitudes in combination with the standard learning rate in Keras. We also find that training the neural network using the mean squared error (MSE) as the loss function gives the highest retrieval accuracies, compared to the cate-

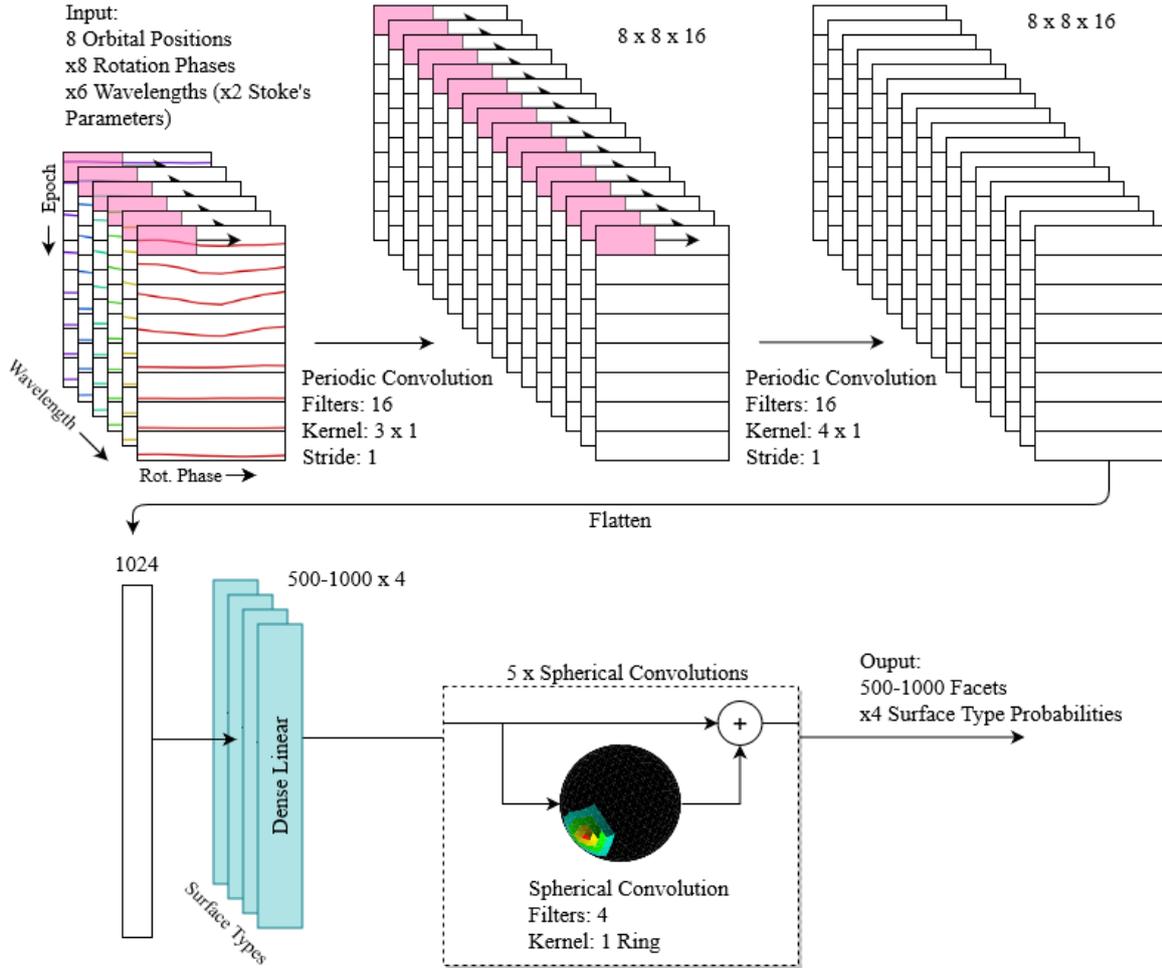


Figure 6.2: The architecture used to classify facets on a planet as one of the four surface types. Several periodic convolutions are used to recognize patterns in the curves similar to the rotation axis retrieval network in Figure 4.3. To maintain a high level of resolution of rotation phases, no down-sampling is used in the convolutions. Four dense layers with 500-1000 nodes estimate the probability of each surface type for all visible facets. 5 spherical convolutions are performed in series and added to the final result. 4 filters are used in the spherical convolutions to match dimensions.

gorical cross entropy which is more commonly used for classification problems (Zhang and Sabuncu [42]). It is unclear to us why this is the case, but it may be due to the ill-defined nature of the exocartography inverse problem or a suboptimal network architecture.

6.2. Retrieval Accuracy

In this section, the retrieval accuracies of the neural network architecture shown in Figure 6.2 are investigated. The retrieval accuracy of each surface type is plotted as a function of the noise level in Figure 6.3. The inclination and rotation axis chosen for this graph are the best combination discussed in Figure 5.2. Since the inclination is 30° , the rainbow feature at $\alpha = 38^\circ$ is visible during two of the eight orbital positions and the ocean glint feature is also prominent at the two orbital positions for which $\alpha = 142^\circ$.

The network performs very poorly when trained on Lambertian light curves and applied to directional light curves. This is further discussed in Section 6.3. The surface type that is classified correctly most often is ocean, due to its unique ocean glint feature as well as its characteristic dark colors. We verify that the high accuracy is (at least) partially due to the ocean glint since the Lambertian neural network applied to Lambertian light curves does not perform as well as the directional neural network applied to directional light curves. A similar effect is also seen for cloudy facets, which also have unique, non-Lambertian phase curves (see Figure 3.5). The vegetation and desert surface types, which are modelled as Lambertian surfaces below the atmosphere, do not see an increase in classification accuracy for the directional light curves, when compared to the Lambertian light curves.

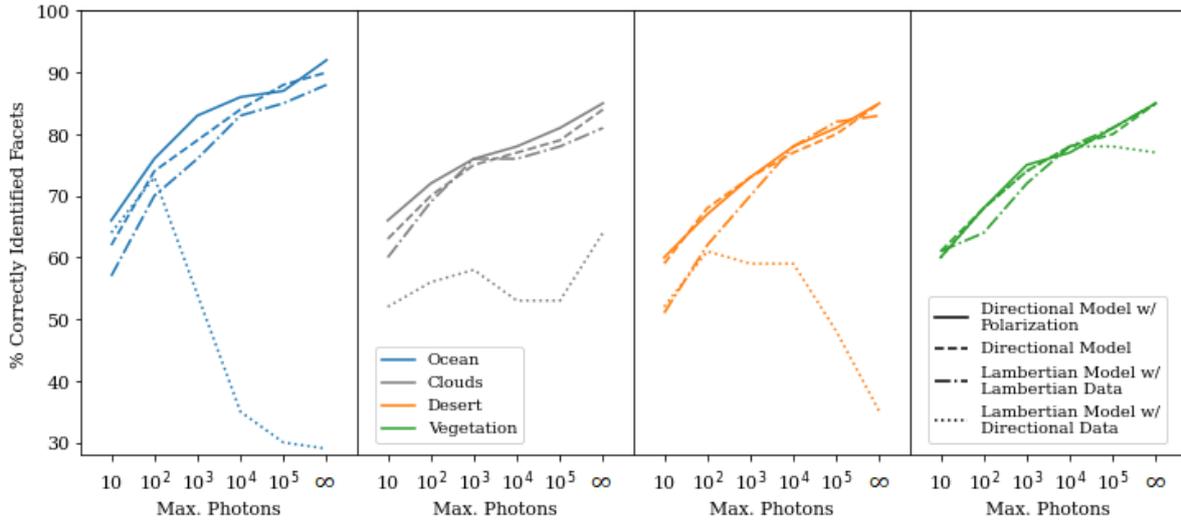


Figure 6.3: The percentage of correctly identified facets in the validation data for different noise levels. Ocean, clouds, desert and vegetation are shown from left to right and have maxima of 92%, 85%, 85% and 85%, respectively. The architecture in Figure 6.2 is trained for the best combination of inclination and rotation axis discussed in Figure 5.2. The directional neural network with polarization is the best in nearly all cases. The Lambertian neural network has a very low classification accuracy when applied to the more accurate directional light curves, except for the vegetation surface type. Ocean has the highest classification accuracy due to its unique glint feature, which is not present in Lambertian light curves. This explains the discrepancy between the Lambertian neural network and directional neural network (a similar effect is seen for cloudy facets).

The retrieved maps of planet Earth are shown in Figure 6.5 as a function of the noise level. These maps have more detail than the albedo maps shown in Figure 5.4, which validates the approach of retrieving surface types rather than albedos. In the absence of noise, all landmasses can be clearly distinguished. At a noise level of $N_{\max} \leq 1,000$ photons North and South America begin to disappear. Finally, for $N_{\max} = 10$ photons only the overall surface type of the planet (vegetation and ocean) is retrieved. Since the level of noise for this case is very high, the specific map of the retrieval can greatly change due to probabilistic noise contributions.

6.3. Validity of Lambertian Assumption

As can be seen in Figure 6.3, the Lambertian neural network yields a very high error rate when applied to directional light curves. As a visual example of this, Figure 6.4 shows the retrieval of the directional light curves of the model Earth by a neural network trained on Lambertian light curves. The neural network incorrectly predicts a very large fraction of facets and predicts bright sandy desert where there should be dark ocean.

Since so far only the ideal geometry from Figure 5.2 is studied, other inclinations are also tested to see if the Lambertian neural network can perform better, as was the case for rotation axis and albedo map retrievals for a face-on orbit. In Figure 6.6, the Lambertian neural network is trained on different inclinations with rotation axes with a tilt near 0° . As was found in Section 4.5 and Section 5.5, the Lambertian assumption is only reliable for face-on orbits and near face-on orbits (at $i = 75^\circ$ all surface types besides ocean have a classification accuracy above 60%). Even for these cases, the directional neural network's classification accuracy is considerably higher and is thus preferred.

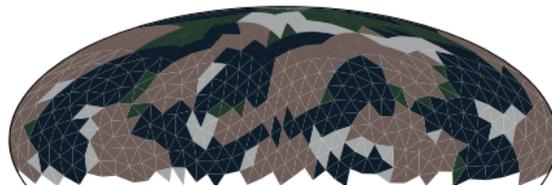


Figure 6.4: The neural network shown in Figure 6.2 is trained with Lambertian light curves and used to retrieve the map of the model Earth (see Figure 6.5) using directional light curves. It does this very poorly due to the inaccuracies of the Lambertian reflection model. The orbital inclination and rotation axis of the planet are the optimal configuration from Figure 5.2.

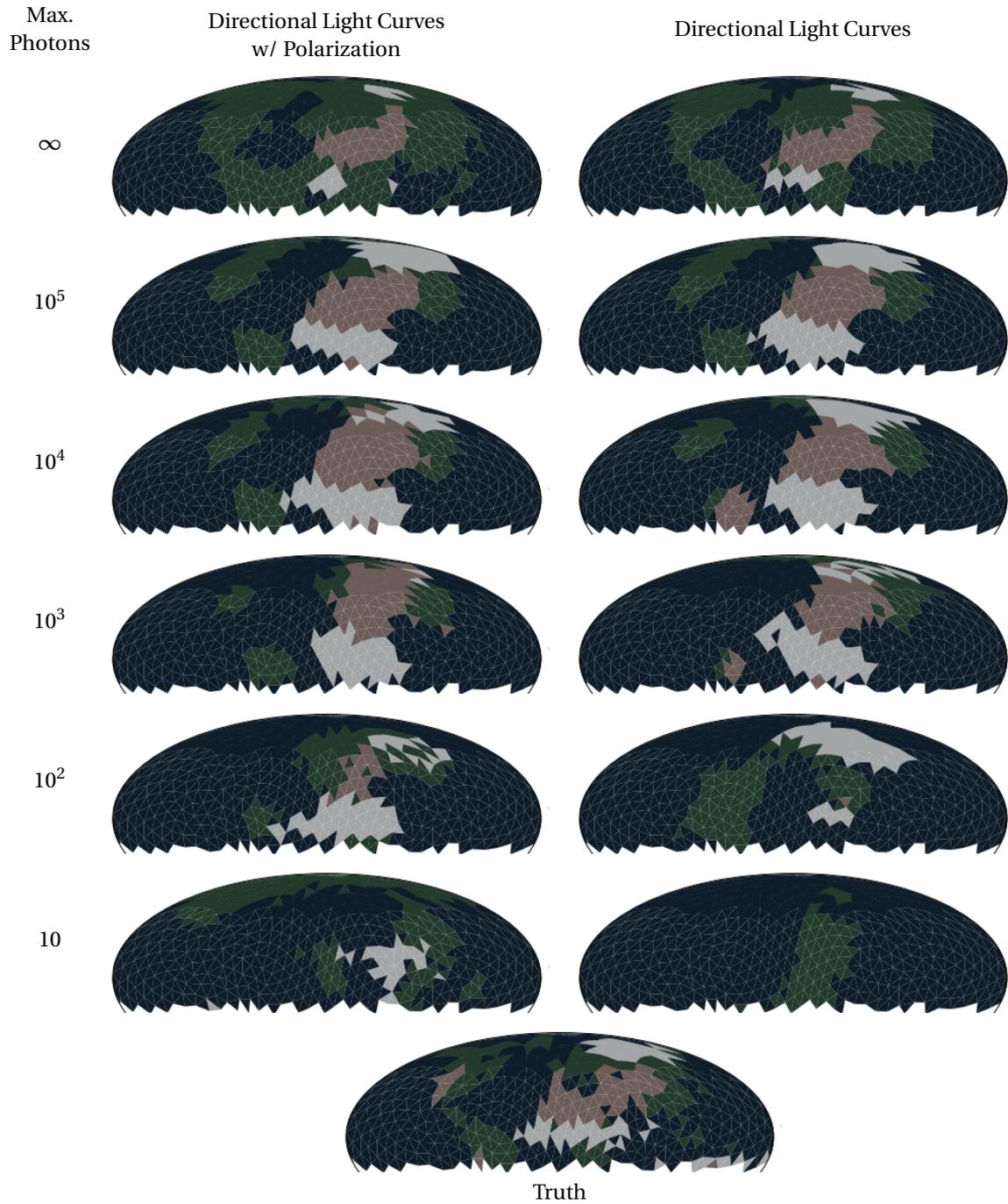


Figure 6.5: Retrievals of Earth for different levels of noise using directional light curves with and without polarization. The architecture from Figure 6.2 is trained for the optimal configuration from Figure 5.2. For $N_{\max} = 10$ photons, the overall surface types of the planet (mostly ocean) are able to be retrieved. At a noise level of $N_{\max} = 1,000$ photons, the Sahara and the Americas are retrieved. At a noise level of $N_{\max} = 100,000$ photons, the rough shapes of the continents and clouds can be distinguished. The classification network provides much more detailed maps than the network for albedo maps shown in Figure 5.4.

The only surface type for which the classification accuracy for the Lambertian neural network is above 65% for all inclinations is vegetation. This may mean that the Lambertian neural network uses the red edge feature of vegetation at the wavelengths of 700 and 800 nm to classify vegetation facets. Since there is very little atmospheric effect at these high wavelengths and because vegetation is modelled as a Lambertian surface, the facets reflect nearly identically for the directional and Lambertian light curves (see the phase curves in Figure 3.5).

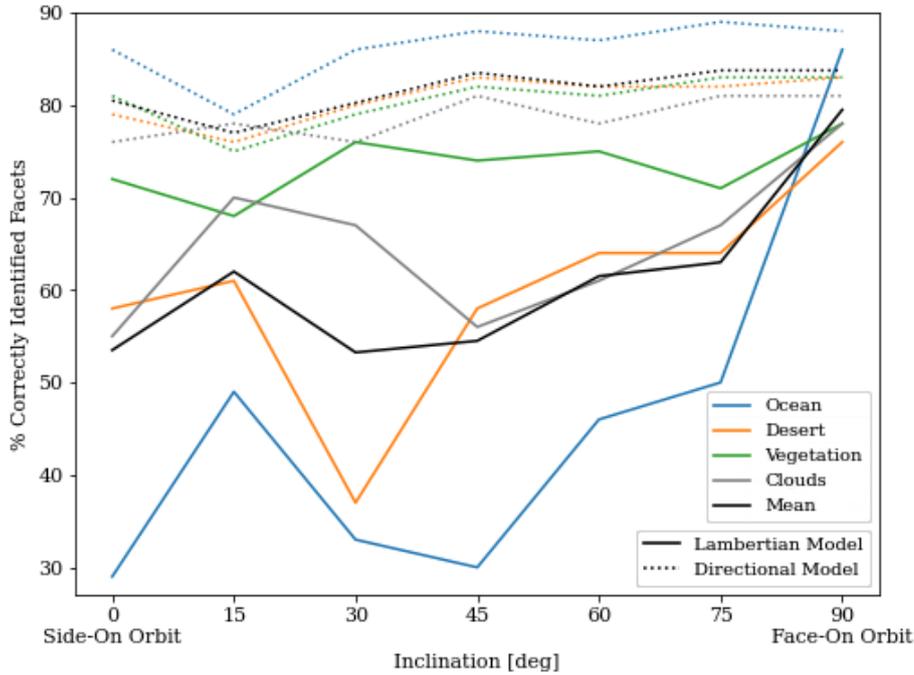


Figure 6.6: The retrieval accuracy of the Lambertian retrieval algorithm applied to directional light curves as a function of inclination. The rotation axes for each inclination are chosen to minimize the difference to the normal of the orbital plane (see Figure 5.5 for examples). For all inclinations besides the face-on orbit, the Lambertian neural network does very poorly. The directional retrieval algorithm (without polarization) is shown in dotted lines for comparison. Even in the face-on orbit where the Lambertian assumption is most valid, the directional neural network has a significantly higher accuracy.

6.4. Benefits of Polarization

To study the effects of including polarization on the retrievals, the confusion matrices are shown for the directional neural networks without and with polarization in Table 6.1 and Table 6.2, respectively. Confusion matrices visualize the performance of classification models by showing the fraction of each actual class (in the columns) that is in each predicted class (in the rows). These show that the retrieval accuracies of the desert and vegetation surface types remain the same after adding polarization as an input to the retrieval. This is to be expected since both are modelled as Lambertian, non-polarizing surfaces. The retrieval accuracy of ocean and cloud facets are increased by 2% and 1%, respectively, due to the unique polarization signature of the ocean glint and rainbow features. This shows that including polarimeters on future telescopes would slightly increase their ability to map exoplanets, especially ones with oceans and clouds.

Pred. \ True	Ocean	Desert	Vegetation	Clouds
Ocean	0.90	0.06	0.03	0.05
Desert	0.05	0.85	0.08	0.06
Vegetation	0.02	0.05	0.85	0.05
Clouds	0.03	0.04	0.04	0.84

Table 6.1: The confusion matrix for the neural network in Figure 6.2 with directional light curves without polarization for planets with optimal configuration from Figure 5.2. Each number shows the fraction of each surface types' facets (columns) that are classified as a specific type (rows). No noise is applied to the curves. The ocean surface type is classified correctly most often due to its unique glint feature and dark colors. The diagonal of the confusion matrix is each surface's accuracy for $N_{\max} = \infty$ (no noise) in Figure 6.3.

Pred. \ True	Ocean	Desert	Vegetation	Clouds
Ocean	0.92	0.06	0.03	0.05
Desert	0.04	0.85	0.08	0.05
Vegetation	0.01	0.05	0.85	0.04
Clouds	0.03	0.03	0.04	0.85

Table 6.2: The same confusion matrix as Table 6.1 but for light curves with polarization. Using polarization increases the retrieval accuracy of ocean facets by 2% and cloudy facets by 1% due to their unique polarization signatures (ocean glint and rainbow). Values that increased or decreased in comparison to Table 6.1 are marked in green and red, respectively.

7

Conclusion

In this thesis, neural networks are used to retrieve exoplanet maps based on variations in reflected star light. Due to the huge distances to other stars, the planets cannot be resolved and are instead observed as unresolved pixels. If the planet is not homogeneous, the pixel's brightness and polarization vary as the planet rotates about its own axis and orbits its star, giving information about longitudinal and latitudinal variations, respectively. By observing the planet at different orbital positions and rotation phases, a map of the planet may be reconstructed from the data.

The first step in retrieving the planet map is determining the planet's rotation axis, which is done in Chapter 4 by the neural network shown in Figure 4.3. We have implemented a new variation of 1D convolutions that takes advantage of the periodic nature of the light curves, leading to 10% better results for rotation axis retrievals. The periodic convolutions are followed by a number of dense layers with PReLU activation functions.

The accuracy of the map predicted by the neural network is inclination-dependent, decreasing from a mean squared error (MSE) of 0.011 for side-on orbits to 0.006 for face-on orbits (the axes have a magnitude of 1). The case of exact side-on observation needs to be treated as a special case, because mirror reflection of the configuration in the orbital plane does not change the signal. This degeneracy has been accounted for in this algorithm. To our knowledge, this degeneracy has not been discussed in publications before. The degeneracy is not problematic as both solutions can be found and they are not fundamentally different types of planets.

In Chapter 5, we show that the albedo maps retrievals by other authors can be roughly replicated using a single-layer neural network with observations as inputs and facet albedos as outputs. The accuracy of the retrieval can be further increased by adding one layer of periodic convolutions and one additional dense layer of neurons resulting in a MSE between 0.014 and 0.022, depending on the combination of inclination and rotation axis. For face-on orbits, the retrievals are most accurate when the tilt of the planets is near 90° and for side-on orbits the retrievals are most accurate when the tilt of the planets is near 0° (there are no seasons when the tilt is 0°). It is also shown that relative, normalized curves such that the maximum flux is equal to one can be used for retrieval by scaling the map with the network architecture shown in Figure 5.3.

In Chapter 6, we test a new approach to planet mapping by predicting surface types rather than albedos. This approach is used since some surface types reflect starlight in a non-Lambertian manner, which is simulated using the method by Rossi et al. [32]. A relatively simple neural network (see Figure 6.3) can create detailed planet maps of this type that predict up to 92% of ocean facets and 85% of other facet types correctly. The network uses periodic convolutions followed by a dense layer that assigns probabilities for each surface type to the facets. For regularization, spherical convolutions that are an improved version of the method by Krachmalnicoff and Tomasi [26] are applied to the map in a ResNet fashion. When applied to a model Earth, the retrieved map shows that the Sahara, Europe, Asia, the Americas and cloud patterns can all be retrieved for a maximum photon number of 1000. Since we neglected all other types of noise, the shot noise alone limits the possibility of the retrieval of the Earth map in such detail by the HabEx telescope in combination with a star shade to a maximum of 75 ly.

In this research, the validity of the Lambertian assumption is assessed. We show that the retrieved rotation axes, albedo maps and surface type maps are all usually poor, except for the case of face-on observation, in which case the signal is mostly Lambertian. The MSE of the rotation axis retrievals is reduced by a factor 10

if directional reflection is taken into account, compared to Lambertian reflection (see Figure 4.9). Retrieving albedo maps using the Lambertian assumption creates concentric patterns about the poles (see Figure 5.5) and retrieving surface type maps also creates erroneous maps (see Figure 6.4). These errors are due to the overestimation and underestimation of the planet's brightness for low and high phase angles, respectively, due to effects like Rayleigh scattering, scattering by water clouds and ocean glint that Lambertian reflection do not take into account (see Figure 3.5).

The usefulness of detecting the polarization of reflected starlight for planet mapping is also assessed in this thesis. We show that shot noise effects the polarized signal of an exoplanet to a greater degree than the total flux, since the degree of polarization is typically only around 20% (see Figure 3.11). Including polarization decreases the MSE of the rotation axis retrievals by around 15% (see Figure 4.9). It also increases the accuracy of surface type maps, specifically the retrieval accuracy of ocean facets and cloudy facets by 2% and 1%, respectively (see Figure 6.3.)

Neural networks have been shown to be a promising candidate for retrievals of exoplanet maps from the reflected starlight. Only 8 orbital locations and 8 rotation phases are needed when using the architectures discussed in this thesis. The next generation of telescopes will be in high demand, so long observation times are not guaranteed. However, the characterization and mapping of a potentially habitable Earth-like planet will probably ensure a serious effort to obtain sufficient signal to noise ratios.

8

Recommendations

Some recommendations for future research are listed below:

- Include ice (this surface type is seen on Earth, Mars, Europa, etc.), different colored deserts (such as the red desert seen on Mars) and/or other colored Lambertian surfaces in the training data.
- Included different atmospheric thicknesses and constituents in the training data and include these parameters in the retrieval.
- Include different clouds such as the sulfuric acid clouds seen on Venus, which exhibit a different rainbow feature than water clouds (Hansen and Hovenier [14]).
- Wavelength dependent starlight can be included in the reflection model. In this thesis the star light is always assumed to have a value of 1, but in reality the intensity of star light is wavelength dependent, depending on the temperature of the star.
- Reflected star light from a moon or several moons could be included in the planetary signal and the presence of moon(s) could be included in the retrieval.
- Simulate the reflected flux of the planet during a full orbit, rather than at 8 locations. Using such a light curve for the retrieval (as is done by many other authors) may require new architectures as the number of observations would drastically increase and thus the number of trainable parameters may become too large for efficient training.
- Dynamic maps with diurnal and seasonal variations in the cloud maps could be used. This would require new neural networks architectures to map the underlying surface in addition to the dynamic cloud map.
- More accurately simulate noise. Effects such as instrumental noise and background noise (due to the atmosphere, (exo-)zodiacal dust, refraction around the star shade, other stars etc.) are not included and may be of greater or similar magnitude than the shot noise.
- Verify the retrieval algorithms with light curves of planets and/or moons in our solar system.
- Try higher and/or lower resolution maps planet maps. The HEALPix pixelization scheme could also be tried instead of a Fibonacci sphere.
- Include different ratios of orbital and rotational period. In this research we assume that the movement of the planet in its orbit is negligible during one rotation about its axis but this is not true for all planets (for example Mercury has a 3:2 spin-orbit resonance).
- The rotation of the planet during one observational integration period causes smearing, which could be included in the computation of the reflected fluxes.

Bibliography

- [1] Bilal Alsallakh, Narine Kokhlikyan, Vivek Miglani, Jun Yuan, and Orion Reblitz-Richardson. Mind the pad – cnns can develop blind spots, 10 2020.
- [2] A. Asensio Ramos and E. Pallé. Planet cartography with neural learned regularization. *Astronomy & Astrophysics*, 646:A4, Feb 2021. ISSN 1432-0746. doi: 10.1051/0004-6361/202040066. URL <http://dx.doi.org/10.1051/0004-6361/202040066>.
- [3] Abdul Badshah, Jamil Ahmad, Nasir Rahim, and Sung Baik. Speech emotion recognition from spectrograms with deep convolutional neural network. pages 1–5, 02 2017. doi: 10.1109/PlatCon.2017.7883728.
- [4] Webster Cash. Detection of earth-like planets around nearby stars using a petal-shaped occulter. *Nature*, 442:51–3, 08 2006. doi: 10.1038/nature04930.
- [5] Courtney Dressing and David Charbonneau. The occurrence of potentially habitable planets orbiting m dwarfs estimated from the full kepler dataset and an empirical measurement of the detection sensitivity. *The Astrophysical Journal*, 807, 01 2015. doi: 10.1088/0004-637X/807/1/45.
- [6] Siteng Fan, Cheng Li, Jia-Zheng Li, Stuart Bartlett, Vijay Natraj, David Crisp, and Yuk Yung. Earth as an exoplanet: A two-dimensional alien map. *Astrophysical Journal Letters*, 882, 09 2019.
- [7] Ben Farr, Will Farr, Nicolas Cowan, Hal Haggard, and Tyler Robinson. Exocartographer: A bayesian framework for mapping exoplanets in reflected light. *The Astronomical Journal*, 156, 02 2018. doi: 10.3847/1538-3881/aad775.
- [8] Yuka Fujii and Hajime Kawahara. Mapping earth analogs from photometric variability: Spin-orbit tomography for planets in inclined orbits. *The Astrophysical Journal*, 755, 04 2012. doi: 10.1088/0004-637X/755/2/101.
- [9] Yuka Fujii, Hajime Kawahara, Yasushi Suto, Atsushi Taruya, Satoru Fukuda, Teruyuki Nakajima, and Edwin Turner. Colors of a second earth: Estimating the fractional areas of ocean, land, and vegetation of earth-like exoplanets. *The Astrophysical Journal*, 715, 04 2010. doi: 10.1088/0004-637X/715/2/866.
- [10] Álvaro González. Measurement of areas on a sphere using fibonacci and latitude–longitude lattices. *Mathematical Geosciences*, 42(1):49–64, 2010. ISSN 1874-8961. doi: 10.1007/s11004-009-9257-x. URL <https://dx.doi.org/10.1007/s11004-009-9257-x>.
- [11] K. M. Gorski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759–771, 2005. ISSN 0004-637X. doi: 10.1086/427976. URL <https://dx.doi.org/10.1086/427976>.
- [12] A. Groot, L. Rossi, Victor Trees, J. Cheung, and D. Stam. Colors of an earth-like exoplanet – temporal flux and polarization signals of the earth. *Astronomy & Astrophysics*, 640, 08 2020. doi: 10.1051/0004-6361/202037569.
- [13] James Hansen and Larry Travis. Light scattering in planetary atmosphere. *Space Science Reviews*, 16: 527–610, 01 1974. doi: 10.1007/BF00168069.
- [14] James E. Hansen and J. W. Hovenier. Interpretation of the Polarization of Venus. *Journal of the Atmospheric Sciences*, 31(4):1137–1160, 05 1974. ISSN 0022-4928. doi: 10.1175/1520-0469(1974)031<1137: IOTPOV>2.0.CO;2. URL [https://doi.org/10.1175/1520-0469\(1974\)031<1137: IOTPOV>2.0.CO;2](https://doi.org/10.1175/1520-0469(1974)031<1137: IOTPOV>2.0.CO;2).

- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *IEEE International Conference on Computer Vision (ICCV 2015)*, 1502, 02 2015. doi: 10.1109/ICCV.2015.123.
- [16] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.
- [17] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2): 251–257, 1991. ISSN 0893-6080. doi: [https://doi.org/10.1016/0893-6080\(91\)90009-T](https://doi.org/10.1016/0893-6080(91)90009-T).
- [18] S. Hunziker, Schmid Hans Martin, D. Mouillet, Julien Milli, Alice Zurlo, Philippe Delorme, L. Abe, Henning Avenhaus, Andrea Baruffolo, A. Bazzon, A. Boccaletti, P. Baudoz, J. Beuzit, Marcel Carbillet, G. Chauvin, R. Claudi, A. Costille, J.-B Daban, S. Desidera, and Fabiano Wildi. Refplanets: Search for reflected light from extrasolar planets with sphere/zimpol. *Astronomy & Astrophysics*, 634, 02 2020.
- [19] JPL. Habex final report. Technical report, NASA, 2019. URL <https://www.jpl.nasa.gov/habex/pdf/HabEx-Final-Report-Public-Release.pdf>.
- [20] Theodora Karalidi, Daphne Stam, and J. Hovenier. Flux and polarisation spectra of water clouds on exoplanets. *Astronomy & Astrophysics*, 530:69–, 06 2011. doi: 10.1051/0004-6361/201116449.
- [21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116, 2020. doi: 10.1109/CVPR42600.2020.00813.
- [22] Hajime Kawahara. Frequency modulation of directly imaged exoplanets: Geometric effect as a probe of planetary obliquity. *The Astrophysical Journal*, 822, 03 2016. doi: 10.3847/0004-637X/822/2/112.
- [23] Hajime Kawahara and Kento Masuda. Bayesian dynamic mapping of an exo-earth from photometric variability. *The Astrophysical Journal*, 900:48, 08 2020. doi: 10.3847/1538-4357/aba95e.
- [24] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.
- [25] Serkan Kiranyaz, Onur Avci, Osama Abdeljaber, Turker Ince, Moncef Gabbouj, and Daniel Inman. 1d convolutional neural networks and applications: A survey, 05 2019.
- [26] N. Krachmalnicoff and M. Tomasi. Convolutional neural networks on the healpix sphere: a pixel-based algorithm and its application to cmb data analysis. *Astronomy & Astrophysics*, 628, 07 2019. doi: 10.1051/0004-6361/201935211.
- [27] Yann Lecun, Bernhard Boser, John Denker, Don Henderson, R. Howard, Wayne Hubbard, and Larry Jackel. Handwritten digit recognition with a back-propagation network. *Neural Information Processing Systems*, 2:396–404, 01 1989.
- [28] Jack Lissauer and Imke Pater. *Fundamental Planetary Science: Physics, Chemistry and Habitability*. 07 2019. ISBN 9781108411981. doi: 10.1017/9781108304061.
- [29] Michel Mayor and Didier Queloz. A jupiter-mass companion to a solar-type star. *Nature*, 378:355–+, 11 1995. doi: 10.1038/378355a0.
- [30] Torben Mogensen. Planet map generation by tetrahedral subdivision. pages 306–318, 01 2010. doi: 10.1007/978-3-642-11486-1_26.
- [31] Aki Roberge, Christine Chen, Rafael Millan-Gabet, Alycia Weinberger, Philip Hinz, Karl Stapelfeldt, Olivier Absil, Marc Kuchner, Geoffrey Bryden, and the Team. The exozodiacal dust problem for direct observations of exoearths. *Publications of the Astronomical Society of the Pacific*, 124, 03 2012. doi: 10.1086/667218.
- [32] Loïc Rossi, Javier Berzosa-Molina, and Daphne Stam. Pymiedap: a python–fortran tool to compute fluxes and polarization signals of (exo)planets. *Astronomy & Astrophysics*, 616, 04 2018. doi: 10.1051/0004-6361/201832859.

- [33] N. Jeremy Kasdin Sara Seager. Starshade rendezvous probe study report. Nasa astrophysics probe study, JPL, Goddard Space Flight Center, MIT, Princeton University, Northrup Grumman, 2018.
- [34] Juergen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61, 04 2014. doi: 10.1016/j.neunet.2014.09.003.
- [35] S. Seager, E. Turner, J. Schafer, and E. Ford. Vegetation’s red edge: a possible spectroscopic biosignature of extraterrestrial plants. *Astrobiology*, 5 3:372–90, 2005.
- [36] D. M. Stam. Spectropolarimetric signatures of earth-like extrasolar planets. *Astronomy & Astrophysics*, 482(3):989–1007, Mar 2008. ISSN 1432-0746. doi: 10.1051/0004-6361:20078358. URL <http://dx.doi.org/10.1051/0004-6361:20078358>.
- [37] Daphne Stam, W Rooij, G Cornet, and J. Hovenier. Integrating polarized light over a planetary disk applied to starlight reflected by extrasolar planets. *Astronomy & Astrophysics*, 452:669–683, 06 2006. doi: 10.1051/0004-6361:20054364.
- [38] Warisa Thangjai and Sa-aat Niwitpong. Confidence intervals for the signal-to-noise ratio and difference of signal-to-noise ratios of log-normal distributions. *Stats*, 2:164–173, 02 2019. doi: 10.3390/stats2010012.
- [39] Victor Trees and D.M. Stam. Blue, white, and red ocean planets. simulations of orbital variations in flux and polarization colors. *Astronomy & Astrophysics*, 626, 04 2019. doi: 10.1051/0004-6361/201935399.
- [40] Mikko Tuomi, H. Jones, J. Barnes, Guillem Anglada-Escudé, and James Jenkins. Bayesian search for low-mass planets around nearby m dwarfs. estimates for occurrence rate based on global detectability statistics. *Monthly Notices of the Royal Astronomical Society*, 441, 03 2014. doi: 10.1093/mnras/stu358.
- [41] Panqu Wang, Pengfei Chen, Ye Yuan, Ding Liu, Zehua Huang, Xiaodi Hou, and Garrison Cottrell. Understanding convolution for semantic segmentation. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 02 2017. doi: 10.1109/WACV.2018.00163.
- [42] Zhilu Zhang and M. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *NeurIPS*, 2018.