

Mapping Saturn using deep learning

I. P. Waldmann^{1,*} & C. A. Griffith²

¹*Department of Physics & Astronomy, University College London, Gower Street,
WC1E6BT London, United Kingdom*

²*University of Arizona, Lunar and Planetary Laboratory, 1629 E University Blvd,
Tucson, AZ 85721-0092*

** Corresponding author*

Nature Astronomy, DOI: 10.1038/s41550-019-0753-8

Clouds and aerosols provide a unique insight into the chemical and physical processes of gas-giant planets. Mapping and characterising the spectral features indicative of the cloud structure and composition, enables an understanding of a planet’s energy budget, chemistry and atmospheric dynamics (e.g. [1, 2, 3, 4]). Current space missions to solar-system planets produce high-quality data sets, yet the sheer amount of data obtained often prohibits detailed ‘by hand’ analyses. Current techniques mainly rely on two approaches: 1) identify the existence of spectral features by dividing fluxes of two or more spectral channels; 2) perform full radiative transfer calculations for individual spectra. The first method suffers from accuracy whilst the second from scalability to the whole planetary surface. Here we developed a deep learning algorithm, PlanetNet, able to quickly and accurately map spatial/spectral features across large, heterogeneous areas of a planet. We demonstrate PlanetNet on Saturn’s 2008 storm[5], enhancing the scope of the area previously studied. Our spectral-component maps indicate compositional and cloud variations of the vast region affected by the storm showing regions of vertical upwelling, and diminished clouds at the centre of compact storms. This analysis quickly and accurately delineates the major components of Saturn’s storm, thereby indicating regions that can be probed deeper with radiative transfer models.

The Visual and Infrared Mapping Spectrometer (VIMS) on the Cassini space-craft [6] is a two-channel mapping spectrometer, which obtains spatially resolved spectra across a 64×64 pixel array. In this work we use the near-infrared channel, which spans 256 contiguously sampled band-passes ranging from 0.85 to 5.1 μm with constant $\delta\lambda = 0.016 \mu\text{m}$. Hence each data cube (or hyperspectral image) contains 4096 individual spectra. The observations of Saturn’s 2008 storm[5] are particularly well suited for this work as they encompass multiple, adjacent storms, providing a complex atmospheric feature space to be analysed by PlanetNet. In particular, the data cube V1581233933 contains a rare ammonia ice feature, detected by [6], which projects a ‘‘S’’ shaped feature on Saturn’s disk. This data cube, along with two spatially adjacent cubes, has previously been analysed by the spectral band division method[5], which allows comparison to our approach. In this study we re-analyse the three original cubes along with three additional adjacent data cubes, all of which were obtained on February 9th 2008.

PlanetNet is capable of non-parametrically identifying faint features in hyperspectral images and once trained on a given feature, able to search for it across highly heterogeneous data sets. The algorithm consists of two parts: 1) a spectral clustering algorithm to identify an initial feature set; 2) a double-stream deep convolutional neural network (CNN). Here we provide a brief overview of the algorithm. Additional details are given in the Method section. Spectral clustering[7] is a versatile clustering algorithm suited for high-dimensional vector spaces. Unlike with more traditional clustering algorithms such as k-nearest neighbours or k-means, spectral clustering is not dependent on the convexity of the individual cluster sets and can identify highly non-convex clusters. We compute the number of clusters in the data and assign a cluster label to each spectrum, as discussed in more detail in the Methods section. Once we obtain our initial clusters, we trim all data within one pixel of a cluster edge to avoid edge uncertainties and proceed to train the neural network (NN) using the remaining spectra. The NN contains two branches, a spatial and a spectral channel. The spectral branch takes each remaining spectrum and trains a two layer CNN using ReLU (rectified linear unit) activation functions and two pooling layers. Surrounding each spectrum, we compute a 20x20 spatial image by averaging the spectral cube along the spectral axis. This is the input for the spatial channel which otherwise follows the same NN architecture as the spectral channel. By analysing both spectral and spatial information, we can take into account the morphological and spectral signatures of atmospheric features on Saturn. In other words, a dark storm, for example, will have a distinct spectrum and spatial morphology that correlate together. By including these spatial-spectral correlations, the neural network will take all possible information available into account. Finally, the output of both spectra and spatial channels is fed into a fully connected layer which links to the cluster labels via a logistic regression layer. Figure 1 shows a schematic of the full network. Similar network designs have been successfully used in aerial image classification of commercial land-usage (e.g. [8, 9]) for a recent review we refer the reader to [10].

We verify the classification accuracies using two methods: 1) During training, 30% of the TC are reserved as ‘test data’. The test data is randomly chosen from the training set and is used to verify the classification accuracy on unseen data during training. Classification on the test data achieves $\sim 90\%$ accuracy. After training, we further test the trained PlanetNet on a resampled version of the TC by rotating the spatial data and interpolating spectral data within each cluster (see Methods for more detail). Here we achieve an accuracy of 93%, consistent with accuracies obtained from the test data.

We apply PlanetNet to the data cube V1581233933, and hereafter refer to this data set as the training cube (TC) in which we identify 5 clearly distinguishable clusters of spectral/spatial features. We confirm the existence of these clusters by performing a principal component analysis (PCA[25]). Whereas PlanetNet clusters spatial/spectral data by their similarity (low statistical distance), PCA decomposes the spectral cube by its variance. Should distinct spectral regions exist, we expect to see signatures of these clusters also in the variance

of the data. PlanetNet clusters are clearly visible in the principal components and we refer the reader to the Method section for a more in-depth discussion.

The left of Figure 2 presents the spatial extent of the TC and a map of the spectra features, while the right hand side shows the spectral characteristics of the 5 spectral clusters identified. Here, the blue region corresponds to a large stormy region (SR) surrounding the central dark storm (purple/green) and label 1 denotes the centre of the “S” feature. Each cluster is distinguished by its absorption and scattering characteristics, indicative of the cloud structure and gas composition. Most salient are the spectral differences between the region surrounding the dark storm features (blue region here-forth referred to as SR) in contrast to the unperturbed regions (red/orange), and the unique signatures of the black storms (purple/green). Spectra 1 & 4 in Figure 2 are examples of spectra belong to the SR region. We find that regions unaffected by the storm (e.g. spectra 2 and 3) display the brightest albedo at 1 - 2 μm (Fig. 2). At these wavelengths, the bands of the well-mixed CH_4 are modulated by clouds, the brightness of which indicates high aerosols. In contrast, the region surrounding the storm features (blue) is dimmer at 1 - 2 μm , suggesting either lower clouds, or, more likely, given the great extent of the SR region, these are spectrally darker clouds, as postulated by [5]. This interpretation indicates that the blue regions encapsulate current and prior storms darkened by the upwelling material of lower albedo.

These SR features, based on their 1 - 2 μm spectra and low 5 μm flux, contain relatively large particles. The brightest of these features (spectrum 1 in Figure 2), forms an “S” feature, which coincides with the position of the electric discharges measured on February 9, 2008 by Cassini’s Radio and Plasma Wave Science (RPWS) instrument[5, 12, 13, 14]. We find the “S” feature to be the proverbial “tip of the iceberg” of a much larger region of upwelling. In Supplementary Figures 1 & 2, we map the L^2 distance (i.e. spectral difference, see equation 7 in Method) between the peak emission of the “S” shaped feature and the remaining SR region. We find that across the SR region the “S” shaped feature contains the highest 1 - 2 μm but lowest 5 μm flux suggesting the particle size to be highest there.

Baines et al.[5] hypothesise that the particles found in the “S” feature are condensed NH_3 as indicated by colour maps of the continuum (0.93 μm), methane (0.90 μm) and the NH_3 ice feature (2.73 μm). Our analysis defines the relative spectrum of the “S” and find that the “S” spectrum displays an absorption feature at roughly 2.74 μm – 2.85 μm . This is part of a broader continuum that resembles the NH_3 ice spectrum characteristic of that observed in Jupiter’s storms[15]. Similarly, the averaged spectrum of the full SR region indicates a similar although weak ammonia ice feature.

Saturn’s thermal flux, measured long-ward of $\sim 4.5 \mu\text{m}$ also differentiates the ambient from the stormy atmospheres. The SR flux at 5 μm exhibits a comparatively low brightness indicative of higher atmospheric opacity at the 5 μm wavelength due to particles roughly of sizes 5 μm and larger. In particular, the PlanetNet algorithm identifies clusters pertaining to Saturn’s dark storm. The associated spectra (5 and 6) indicate notable spectral structure. Both

spectra have 1 - 2 μm albedos darker than that of the other regions. Relative to the mean storm region (SR), spectra of the dark storms display absorption at 4.5 - 4.9 μm characteristic of PH_3 and consistent with the visibility of this feature as indicated in radiative transfer (RT) models[16]. PH_3 is a condensible species, the abundance of which increases by vertical upwelling at the observable levels. The enhancement of this species indicates that the storm experiences strong vertical updrafts. Saturn’s emission, longward of 4.5 μm , differentiates the centre of the storm (Spectrum 5) from its immediate surroundings (Spectrum 6). The storm centre exhibits a strong flux, indicating the clearest view into the hotter and deep atmosphere, the clarity of which suggests the eye of a storm. In contrast, the 5 μm flux surrounding the storm is commensurate with that of the other clusters.

Supplementary Fig. 1 shows the identified storm region in more detail and map the statistical distance (equation (7) in Method) between the peak emission of the “S” shape feature (marked as 1) and all other spectra in the SR labelled regions. Similar to Figure 2 we plot individual spectra scattered across the SR region and show the spectral differences to the “S” feature by plotting their differences. Whereas most spectral bands are invariant, we do find for all spectra that the 1 - 2 μm albedo is suppressed and the thermal emission increased compared to the “S” shape feature. This furthermore corroborates the conclusion that the “S” shaped feature is indeed the peak of a larger upwelling. In Supplementary Fig. 3 we show the same statistical distance map as in Supplementary Fig. 1 but taking the centre of the dark storm (point 5 in Figure 2) as the reference point. This figure hence shows the likeness of spectra compared to the central dark storm signature. The SR region stands out as being distinct from the dark storm signatures, a feature that is further confirmed by the principal component analysis in Supplementary Figures 4 - 6 discussed in the Method section.

Once trained on a hyperspectral cube, PlanetNet allows us to quickly and accurately map salient spectral regions over multiple heterogeneous data sets, spanning a large area of the planet. Whereas our analysis focused on the training cube so far, we now use PlanetNet to map a much larger region by including 5 additional data cubes that encompass the original storm, as well as two smaller storms eastwards. As shown in Figures 3 & 4 we detect the presence of the SR feature, significantly beyond the spatial coverage initially reported by [5]. In addition, we find similar spatial/spectral signatures of the stormy region around a smaller dark storm to the east (42°W), indicating that areas of upwelling are common around dark storms on Saturn. We provide individual maps of each mapped component in the Supplementary material (Supplementary Figures 7-10) as well as the statistical distance map of the SR region across all five data sets in Supplementary Fig. 2.

Past and current planetary missions produce a wealth of data, too abundant to be analysed by “hand”. More traditional data analysis techniques force us to consider only small volumes of data and a global understanding of spatial distributions of spectroscopic features (e.g. clouds on gas giants) is often lost. Maps produced by PlanetNet can give us insight into large-scale dynamics of a

planet, while identifying regions of interest for more traditional radiative transfer calculations. This technique is significantly more sensitive and robust than simple spectral band subtraction or division and can reveal previously unseen dynamics in the atmospheres of giant planets. The ability to identify features in data sets markedly different to the training data (both in pixel scale and observed angle) allows this technique to be easily scalable to large, planet-wide mapping of spectral features. PlanetNet can easily be adapted to other data sets and missions, making it a potentially invaluable tool in the global analysis of planetary mission data in the future.

All correspondences should be addressed to I.P.W (ingo@star.ucl.ac.uk).

Acknowledgements The authors would like to thank the editor and the referees for helping to greatly improve the clarity of this paper.

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 758892, ExoAI) and under the European Union’s Seventh Framework Programme (FP7/2007-2013)/ ERC grant agreement numbers 617119 (ExoLights). IPW furthermore acknowledges funding by the Science and Technology Funding Council (STFC) grants: ST/K502406/1 and ST/P000282/1 and support from Microsoft Azure for Research cloud computing. CAG is funded by the University of Arizona.

Author Contribution I.P.W developed the PlanetNet algorithm and conducted the data analysis. C.A.G. provided the calibrated Cassini/VIMS data and lead the spectral interpretation of the data. Both authors equally contributed to writing the text.

Data Availability The data analysed in this work are available through the Planetary Data System (<https://pds.nasa.gov>). PlanetNet is publicly available through the UCL-Exoplanets GitHub page (<https://github.com/ucl-exoplanets/>). In addition, both code and training data used are permanently archived and can be accessed with the permanent link: <https://osf.io/htgrn> or the Digital Object Identifier: DOI 10.17605/OSF.IO/HTGRN.

References

- [1] Fouchet, T. *et al.* Stratospheric aftermath of the 2010 Storm on Saturn as observed by the TEXES instrument. I. Temperature structure. *Icarus* **277**, 196–214 (2016).
- [2] Fletcher, L. N. *et al.* Thermal Structure and Dynamics of Saturn’s Northern Springtime Disturbance. *Science* **332**, 1413 (2011).
- [3] Barstow, J. K., Irwin, P. G. J., Fletcher, L. N., Giles, R. S. & Merlet, C. Probing Saturn’s tropospheric cloud with Cassini/VIMS. *Icarus* **271**, 400–417 (2016).

- [4] Sánchez-Lavega, A. *et al.* An enduring rapidly moving storm as a guide to Saturn’s Equatorial jet’s complex structure. *Nature Communications* **7**, 13262 (2016).
- [5] Baines, K. H. *et al.* Storm clouds on Saturn: Lightning-induced chemistry and associated materials consistent with Cassini/VIMS spectra. *Planetary and Space Science* **57**, 1650–1658 (2009).
- [6] Brown, R. H. *et al.* The Cassini Visual And Infrared Mapping Spectrometer (Vims) Investigation. *Space Science Reviews* **115**, 111–168 (2004).
- [7] Yu, S. X. & Shi, J. Multiclass spectral clustering. In *Proceedings Ninth IEEE International Conference on Computer Vision*, 313–319 vol.1 (2003).
- [8] Wu, H. & Prasad, S. Convolutional recurrent neural networks for hyperspectral data classification. *Remote Sensing* **9**, 298 (2017).
- [9] Yang, J., Zhao, Y., Chan, J. C. W. & Yi, C. Hyperspectral image classification using two-channel deep convolutional neural network. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 5079–5082 (2016).
- [10] Zhu, X. X. *et al.* Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine* **5**, 8–36 (2017).
- [11] Jolliffe, I. *Principal component analysis* (Springer Verlag, New York, 2002).
- [12] Gurnett, D. A. *et al.* Radio and Plasma Wave Observations at Saturn from Cassini’s Approach and First Orbit. *Science* **307**, 1255–1259 (2005).
- [13] Fischer, G. *et al.* Analysis of a giant lightning storm on Saturn. *Icarus* **190**, 528–544 (2007).
- [14] Fischer, G. *et al.* Atmospheric Electricity at Saturn. *Space Science Reviews* **137**, 271–285 (2008).
- [15] Baines, K. H., Carlson, R. W. & Kamp, L. W. Fresh Ammonia Ice Clouds in Jupiter. I. Spectroscopic Identification, Spatial Distribution, and Dynamical Implications. *Icarus* **159**, 74–94 (2002).
- [16] Sromovsky, L. A., Baines, K. H. & Fry, P. M. Saturn’s Great Storm of 2010-2011: Evidence for ammonia and water ices from analysis of VIMS spectra. *Icarus* **226**, 402–418 (2013).

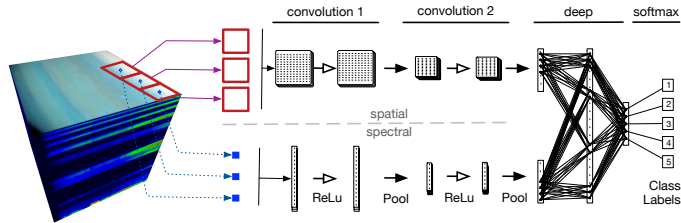


Figure 1: **Flowchart of the PlanetNet algorithm.** The blue dots indicate the central pixel at which the spectrum is extracted from the Cassini/VIMS data cube. The red squares indicate the corresponding spatial patches centred on the central spectral pixels. The spatial and spectral data is fed into two convolutional neural networks for the spatial (top network) and spectral (bottom network) information respectively. Both convolutional networks are linked to a fully connected layer combining spatial and spectral convolutional outputs. The output of the fully connected layer is mapped to the class labels. Please see the Method section for more information.

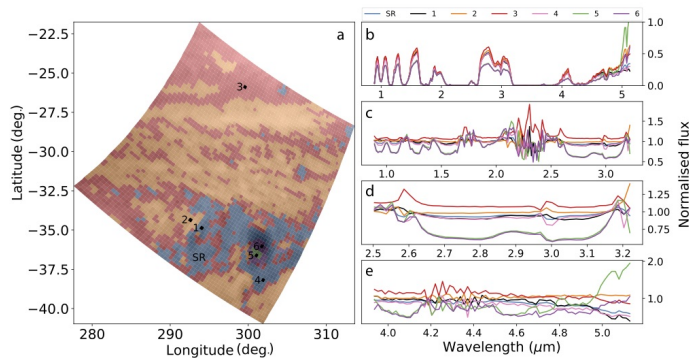


Figure 2: **Spatial and spectral characteristics of the PlanetNet identified features.** A: Map of Cassini/VIMS cube V1581233933, coloured according to the 5 different clusters described in the main text: SR (blue) , 2 (orange), 3 (red), 4 (green), 5 (purple). B: Atmospheric spectra at locations marked on the map (A), spectrum 1 corresponds to the “S” feature of NH_3 ice clouds. The SR spectrum and label correspond to the mean spectrum of the blue designated area on the map. C & D: Same as B above but with the map’s mean spectrum subtracted.

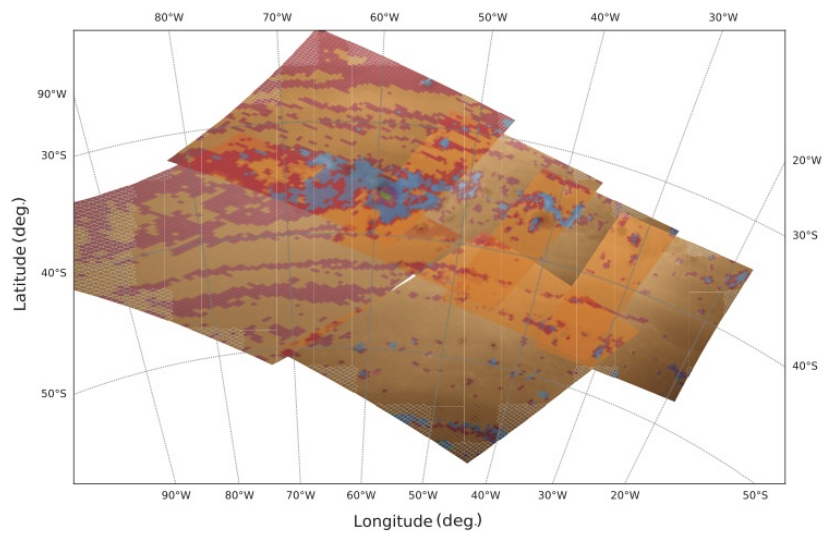


Figure 3: **Cloud distribution as mapped by PlanetNet across six overlapping data sets.** Colours are identical to Figure 2. It is apparent that the SR feature (blue) occurs in the vicinity of dark storms.

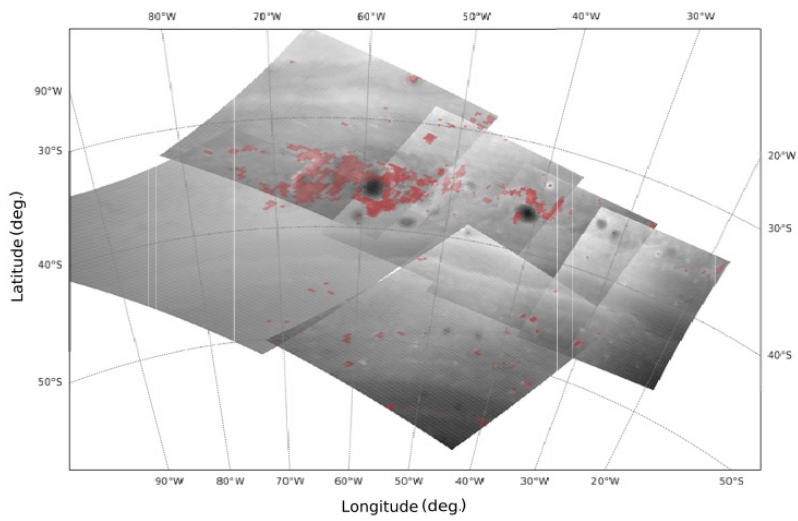


Figure 4: **Stormy region as mapped by PlanetNet across six overlapping data sets.** Same as Figure 3 but only showing the 'stormy region' (SR) referred to in the main text and Figure 2

Methods Here we outline the architecture and training of PlanetNet shown in Figure 1. It is a two-stream convolutional neural network (CNN), which analyses the spatial and spectral data of the VIMS data cube separately before combining information in a common, fully connected network. Convolutional Neural Networks, also known as translation invariant networks, have been designed specifically for image analysis. As opposed to fully connected deep belief architectures, they are translation invariant, meaning that image features can equally be recognised no matter where they occur in the image. Each CNN layer contains three separate stages: 1) Convolution, 2) non-linear transform (ReLU) and 3) down-sampling (pooling). In the convolution stage, the 2D image is convolved with a set of ‘filter functions’ or kernels. These filter functions are continuously learned from the data during training and provide the decomposition into spatial features (e.g. edges, blobs, etc.). We now convolve each filter (computing the dot product) with the input image to obtain the activation map for each filter. In the second stage, we apply a non-linearity transformation to the activation maps. It is now standard practice to use rectifier linear units (ReLUs) or leaky ReLU as CNN non-linearity. We here use the classical ReLU which is given as $f(x) = \max(0, x)$ and effectively removes all negative entries of the activation map, increasing the non-linearity of the transform. Finally, we down-sample in size, or ‘pool’, the activation maps. Here we use simple 2×2 max-pooling, where the maximum value of a 2×2 grid is retained. This procedure is now repeated for the second CNN layer.

The CNN presented here is classical in all aspects apart from the inclusion of two separate and simultaneous streams that treat the spatial and spectral data independently before combining both streams in a fully connected layer. Due to the intrinsic correlations in the spectral channel, we opt for a 1D CNN architecture instead of fully connected layers only. We refer the interested reader to the standard literature on neural networks and CNNs (e.g. [17, 18, 19]). The neural network has been implemented in Tensorflow[20]. The input data on which training and subsequent classification is performed is denoted by

$$\mathbf{x} \in \mathbb{R}^{H^0 \times W^0 \times S^0} \quad (1)$$

where \mathbf{x} is a three-tensor of dimensions H^0 , W^0 and S^0 , denoting the height, width and spectral axes of the data cube. The zero index denotes the input (i.e. data) layer of the neural network. Here and throughout we define tensors and matrices by lower and upper-case bold letters respectively. For ease of notation, we also define the sub-tensors $\mathbf{x}_\lambda = \mathbf{x}^s$ and $\mathbf{x}_\phi = \mathbf{x}^{(H \times W)}$ denoting the spectral and spatial information only. As shown in Figure 1, the CNN features two convolution and pooling layers ($l \in 1, 2, \dots, L$) for each spatial and spectral channel. These can be defined as follows

$$\mathbf{y}_i^l = \text{pool}_P \left[\sum_j^{N^{l-1}} \beta(\mathbf{y}_j^{l-1} \otimes \mathbf{w}_{i,j}^l + \mathbf{b}_i^l) \right] \quad (2)$$

where \mathbf{y}_i^l is the CNN output at layer l and the feature map i . For the spatial

and spectral channels, \mathbf{y}_i^l constitutes a 2D and 3D tensor ($\mathbf{y}_{\lambda,i}^l \in \mathbb{R}^{S^l \times B^l}$ and $\mathbf{y}_{\phi,i}^l \in \mathbb{R}^{H^l \times W^l \times B^l}$) respectively with B denoting the individual training batch size. \mathbf{y}_j^{l-1} is the feature map of the previous layer and $\mathbf{w}_{i,j}^l$ the corresponding kernel of size K^l , and \otimes designates linear convolution. We define the biases as \mathbf{b}_i^l and note that $\mathbf{y}^0 = \mathbf{x}$. The activation function, β , is defined as a linear rectifier (ReLU) unit and a max-pooling has been used for both layers and channels.

After both convolution layers, the spatial and spectral channels are combined to form the input of a fully connected network. We define the fully connected part of the neural network as follows

$$\mathbf{z} = \beta\{\mathbf{W}^c \cdot \beta[\mathbf{W}^{c-1} \cdot (\mathbf{y}_\phi^L \odot \mathbf{y}_\lambda^L) + \mathbf{b}^{c-1}] + \mathbf{b}^c\} \quad (3)$$

where \odot denotes the concatenation of \mathbf{y}_ϕ^L and \mathbf{y}_λ^L , the outputs of the final convolution layers. The fully connected layer indices are given by $c \in 1, 2, \dots, C$ and \mathbf{W} and \mathbf{b} are the weight matrices and biases respectively. The number of neurons per layer is defined by the hyperparameter M^c . For simplicity, we refer to the set of free parameters as $\vartheta = \{\mathbf{w}^l, \mathbf{b}^l, \mathbf{W}^c, \mathbf{b}^c\}$.

Finally, we map \mathbf{z} to our binary training labels, θ , using a softmax regression layer. To train, we now minimise the cross-entropy of the system.

$$Cost = \frac{-1}{n} \sum_x [\theta \ln(\mathbf{a}) + (1 - \theta) \ln(1 - \mathbf{a})] \quad (4)$$

where n is the total number of training data (or batch size), θ the vector of training labels and $\mathbf{a} = \varsigma(\mathbf{z})$, where $\varsigma(\cdot)$ is a sigmoid activation function.

Each Cassini/VIMS data cube consists of two spatial and one spectral axis. The spatial dimensions of the cube are 64×64 pixels and 256 wavelength points. As input to PlanetNet, we require spectra/spatial pairs. These are generated as follows: For every pixel, the full spectrum is extracted, i.e. $S^0 = 256$. Surrounding the central pixel, we also extract the spatial information. The patch size is defined by H^0 and W^0 (which we later determined to be 20 pixels each). Here the spatial patch is the mean average over all wavelengths. We extract spatial/spectral pairs for all 4096 pixels. Note that spatial patches are overlapping from adjacent pixels.

In order to train the neural network, we must obtain labeled data. This can be obtained in two ways: 1) labelling individual pixels by hand to mark predominant features or 2) use a statistical clustering algorithm to obtain estimated labels on which the network can learn and improve. Here we use spectral clustering on the data cube initially presented by [5] to obtain the NH_3 signature. Spectral clustering is a preferable clustering algorithm (compared to for example k-means or nearest neighbours) when the structures of clusters are highly non-convex and simple affinity based discriminators break down. For spectral clustering, we follow [21, 22] and use the sklearn[23] implementation. We define the symmetric graph Laplacian,

$$\mathcal{L}_{sym} = \mathbf{D}^{-1/2} \mathcal{L} \mathbf{D}^{-1/2} \quad (5)$$

where $\mathcal{L} = \mathbf{D} - \mathbf{A}$, \mathbf{A} is the affinity matrix and \mathbf{D} is the degree matrix. The elements of the affinity matrix are computed using

$$a_{i,j} = \exp\left(-\frac{1}{2\sigma^2}d^2[\mathbf{x}_{\lambda,i}, \mathbf{x}_{\lambda,j}]\right) \quad (6)$$

where i, j are the spatial indices of the data and d^2 is the L^2 norm or Euclidian distance given by

$$d_{i,j} = \sqrt{\sum (\mathbf{x}_i - \mathbf{x}_j)^2} \quad (7)$$

The degree matrix is given by

$$\mathbf{D} = \text{diag}\left(\sum_j a_{i,j}\right). \quad (8)$$

Once the graph Laplacian is calculated, we decompose \mathcal{L}_{sym} into its eigenvalues and eigenvectors and rank them from smallest to largest eigenvalue. We estimate the likely number of clusters present in the data by using the eigenvalue heuristic[24]

$$N_{cluster} = \max(\epsilon_{i+1} - \epsilon_i) \quad \text{for } i = [1, 2, \dots, N_\epsilon - 1] \quad (9)$$

We now use k-means clustering on the matrix of eigenvectors to divide the data into $N_{cluster}$ labeled partitions. This provides us with training labels for each spatial coordinate (i, j) .

Since these cluster labels are estimated, some uncertainty in the cluster edges may persist. To mitigate cluster-edge uncertainties, we prune the edges and discard any pixel that lie along a cluster boundary. Finally, the remaining data-label pairs are split into 70% training and 30% validation sets.

The neural network is trained for 2×10^4 iterations with mini-batch sizes of 100 spectral/spatial pairs at a time. We train on the training set only and use the validation set as unseen data to check the network’s ability to generalise over new data and to gauge any over-fitting. Over-fitting would generally be observed in systems with decreasing cross-entropy but static or decreasing validation accuracy. To mitigate over-fitting, we employ a 30% dropout rate across all free variables, ϑ , and a relatively slow learning rate of 0.001. We find that convergence in cross-entropy is usually observed around 1.5×10^4 training steps or less with a monotonically decreasing cross-entropy and increasing validation set accuracy, indicating good convergence and no over-fitting. Training takes 60 minutes on a 6-core Intel Xeon E5 (3.5Ghz) CPU or ~ 5 minutes on a Nvidia Tesla K40 GPU.

In order to optimise the size of the neural network, we ran a grid of 1000 hyperparameter sets. We define the hyperparameters as: the size of the spatial-patch surrounding the central pixel (red square in Figure 1, H_0 and W_0); The number of feature maps for both spatial and spectral channels (N_ϕ^l and N_λ^l), the kernel size (K_ϕ^l and K_λ^l), the down-sampling pool size (P_ϕ and P_λ) and the

layer sized of the fully-connected network (M^c). We define a symmetric square of 20×20 pixels (H^0, W^0) as input and $N^1 = 15$ and $N^2 = 40$ feature maps for both spatial and spectral channels for the first and second layers respectively. Both layers have kernel sizes of $K_\phi^l = 4 \times 4$ and $K_\lambda^l = 4 \times 1$ for the spatial and spectral channels respectively. We use a factor 2 down-sampling with pooling sizes $P_\phi = 2 \times 2$ and $P_\lambda = 2 \times 1$. We summarise the hyperparameter values in Supplementary Table 1.

We investigate the training and prediction accuracies after PlanetNet has been trained on the training cube (TC). Before training, the TC was sub-divided into 70% training data and 30% test data. The test data is not shown to the algorithm during training and is used to independently verify the classification accuracy and provide a diagnostic against over-training. In the case of over-training, the classification accuracy of the training set will steadily increase over the training period, whereas the test set accuracy remains unchanged or decreases. In this case, the neural network is memorising the training set rather than learning how to generalise. Supplementary Fig. 11 shows the training (red) and test (purple) accuracies as function of training duration. Both curves are monotonically increasing and show no signs of overfitting. The training and test sets reach $\sim 95\%$ and $\sim 90\%$ classification accuracies. Furthermore, the same figure plots the cross-entropy (loss function) of PlanetNet during training.

After PlanetNet has been trained, we further verify the classification accuracy by testing the NN on a re-sampled, “synthetic” data set. We refer to this data set as re-sampled test cube (RTC). The RTC was generated by re-sampling the training cube in such a way as to preserve the statistical properties of the data but to appear as a new, unseen data to the NN. First, the TC was divided according to the labels obtained by the spectral clustering. Following the previous notation, we refer to the TC data pertaining to a label as \mathbf{x}_θ , where θ is the index of the label. Within each label subset, we randomised the order of each element of \mathbf{x}_θ to break any close spatial connections in the data. Then we re-sample each train set, both in the spatial \mathbf{x}_θ, ϕ and spectral $\mathbf{x}_\theta, \lambda$ dimensions. For the spatial dimension, we transpose \mathbf{x}_θ, ϕ , which is equivalent to a 90 degree rotation. This rotation preserves the underlying properties of the data but appears as new, unseen data to the NN. The spectral information of each element in $\mathbf{x}_\theta, \lambda$ was replaced with the average of the two adjacent spectra in the RTC. Given that all spectral positions in the RTC are randomised, the newly generated spectra will effectively sample from the distribution of the cluster.

We now let PlanetNet detect cluster labels on RTC using spectral and spatial information, and additionally with spectral or spatial information only. Supplementary Fig. 12 shows the RTC classification for spatial and spectral data. The left shows the ‘ground-truth’ labels, middle the predicted labels by PlanetNet and right is the error matrix of pixels miss-classified. PlanetNet achieves a 93% classification accuracy, which is in agreement with the training/test data classification accuracies.

We now run the classification for spatial-only and spectra-only cases by setting $\mathbf{x}_\theta, \lambda = 0$ and $\mathbf{x}_\theta, \phi = 0$ respectively. This results in very poor cluster label recognition by PlanetNet (48% and 32%, Supplementary Figures 13 & 14,

respectively). However we note that this is to be expected for models that are trained on both, spatial and spectral pairs. Suppressing half of the network’s information results in erroneous predictions. Hence, if the user wants to use, say, spectral data only for their classification, PlanetNet should be re-trained on spectral-only data.

In addition to our analysis using PlanetNet, we have also conducted a classical principal component analysis[25] of the training cube to further verify the classification derived by PlanetNet. In a recent analysis, Griffith et al.[26] has shown that PCA analysis of hyperspectral images of Titan can unveil structural information that otherwise remains hidden in spectral noise. PCA is complementary to the main analysis using PlanetNet as principal components probe a different statistics of the data and can hence verify the veracity of PlanetNet’s analysis. Whereas the spectral clustering of PlanetNet identifies clusters of similar spatial/spectral regions, PCA decomposes the data along the axes of highest variances. In other words, PlanetNet is sensitive to the statistical distance (equation (7)) between data points and hence clusters the data according to their ‘likeness’, whereas PCA on the other hand decomposes the data by its variance, broadly speaking its ‘dissimilarity’. Should distinct spectral classes exist, they will contribute to the variance of the data cube and should hence be separable as individual principal components. Both methods should converge to finding the same regions. We find this to be the case, verifying the validity of our decomposition of the training cube. In Supplementary Figure 12 we show the first four principal components of the 0.88 - 1.66 μm spectral region alone. These regions probe the albedo of the clouds and are hence most sensitive to the high altitude atmospheric variations. The “S” shaped NH_3 ice feature[5] is clearly visible in the second to fourth component. Supplementary Figure 13 shows additional PCA maps of the full wavelength range and the thermal emission component of the spectrum ($>4 \mu\text{m}$) is shown in Supplementary Figure 14. Both maps clearly show the presence of the SR region in their second principal components. When considering the full wavelength range (Supplementary Figure 13) we see the dark storm to be most prominent in the 1st component, whereas Supplementary Figure 14 shows the SR region prominently both in the first and second components. This suggests that spectral variance in the thermal emission part of the spectrum is strongly driven by the SR region and indicative of turbulent upwelling. Future work will explore the addition of component separation methods to further augment feature detection sensitivities.

References

- [17] Bishop, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)* (Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006).
- [18] Bengio, Y. *Learning Deep Architectures for AI*, vol. 2 (Now Publishers Inc, 2009).

- [19] Goodfellow, I., Bengio, Y. & Courville, A. *Deep Learning* (MIT Press, 2016). <http://www.deeplearningbook.org>.
- [20] Abadi, M. *et al.* TensorFlow: Large-scale machine learning on heterogeneous systems (2015). URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- [21] Yu, S. X. & Shi, J. Multiclass spectral clustering. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2, ICCV '03*, 313– (IEEE Computer Society, Washington, DC, USA, 2003). URL <http://dl.acm.org/citation.cfm?id=946247.946658>.
- [22] Ng, A. Y., Jordan, M. I. & Weiss, Y. On spectral clustering: Analysis and an algorithm. In *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*, 849–856 (MIT Press, 2001).
- [23] Pedregosa, F. *et al.* Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011).
- [24] Luxburg, U. A tutorial on spectral clustering. *Statistics and Computing* **17**, 395–416 (2007).
- [25] Jolliffe, I. *Principal component analysis* (Springer Verlag, New York, 2002).
- [26] Griffith, C. A. *et al.* A corridor of exposed ice-rich bedrock across Titan's tropical region *Nat. Astr.* (accepted).

Supplementary Information

Euclidean Distance maps

Additional maps showing individual components as recognised by PlanetNet as well as statistical distance plots. For more information, please refer to the Method section of the main text.

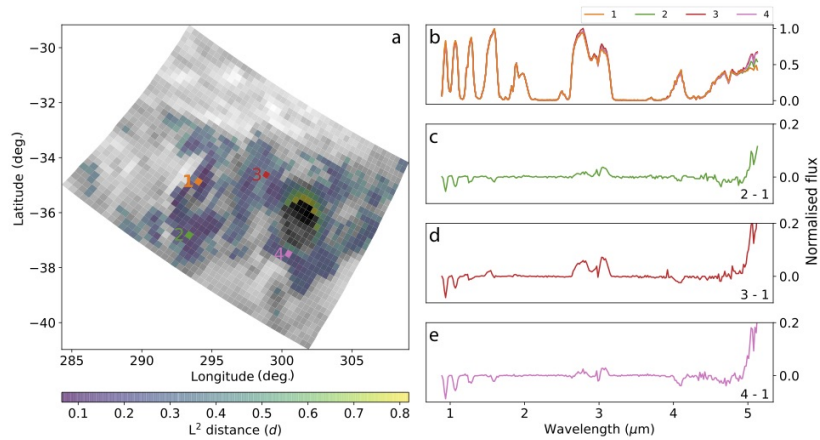


Figure 5: **Map of spectral differences inside the storm region (SR).** A: Map of the SR as function of Euclidian distance (L^2 norm) from the peak emission of the “S” shaped ammonia ice feature, here labelled 1 (orange). The map shows the statistical similarity of each spectrum compared to (1) with low values being more similar and high values being less similar. B: Spectra corresponding to the pixels labelled on the map. C - E: Spectral differences between spectra (2), (3), (4) and (1) respectively.

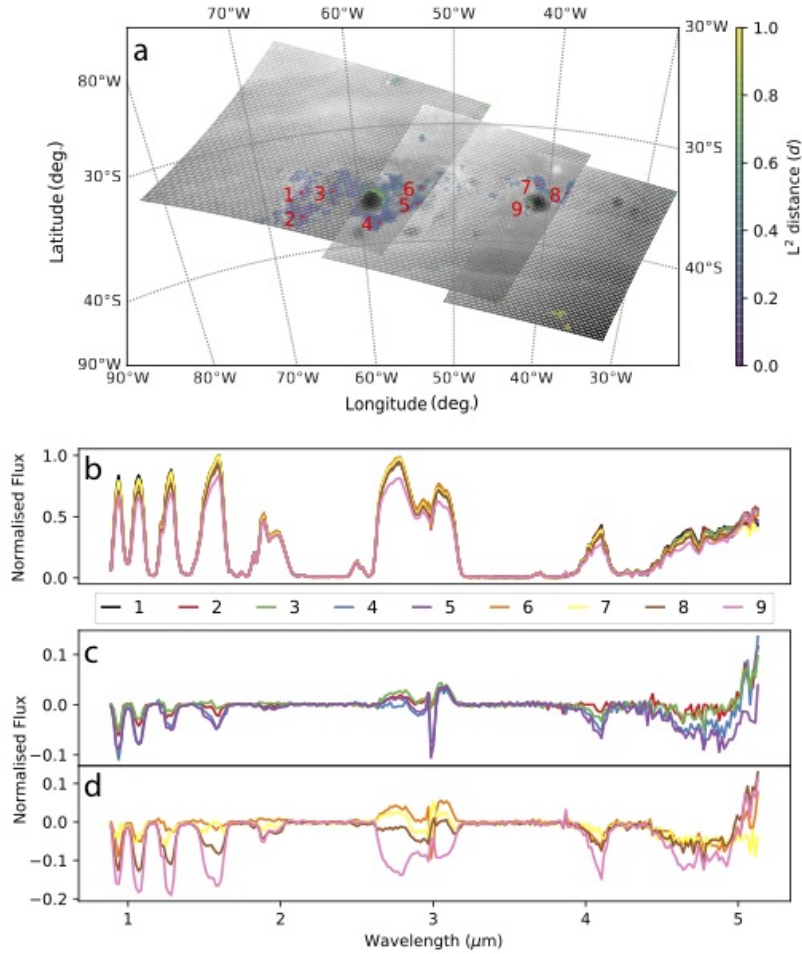


Figure 6: **Map of spectral differences inside the storm region (SR) across all six data sets.** A: Same Euclidean distance map to figure 5 but for the larger area shown in figure 4 of the main text, encompassing the stormy region of the smaller storm at 42°W. All Euclidian distances measured are relative to the spectrum marked as (1), with small distances indicating a higher degree of similarity. B: Spectra pertaining to the pixels marked in the map above. C & D: Spectral differences between spectra in the middle plot and spectrum (1).

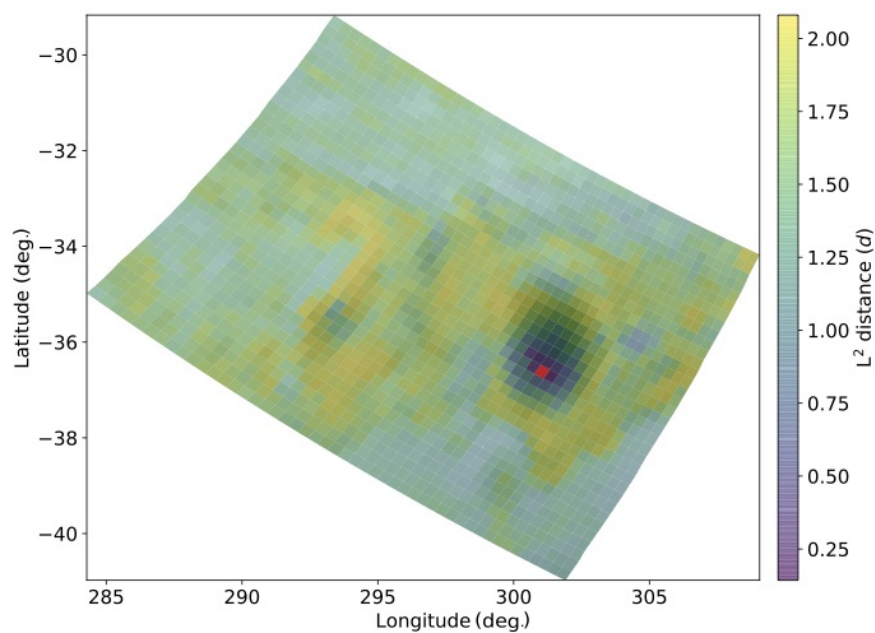


Figure 7: **Map of the Euclidian distance (L^2 norm) from the centre of the dark storm (red pixel) for all spectra (independent of cluster label).** Similar to supplementary figures 5 & 6 the smaller the distance the higher the similarity between two spectra. The SR region surrounding the dark storm stands out as being statistically distant (i.e. dissimilar) from dark storm signatures.

Principal Component Maps

Maps showing the results of the Principal Component Analysis (PCA) explained in the Method section of the main text.

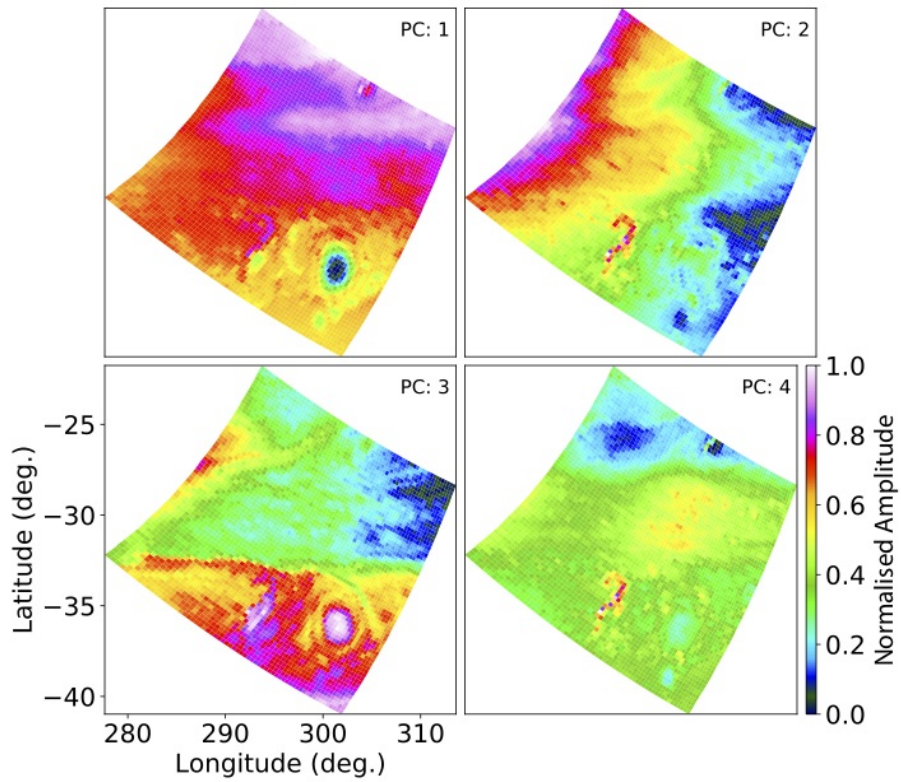


Figure 8: **Spatial map of the principal component analysis of the 0.88 - 1.66 μm spectral region.** Shown are the first four strongest principal components (PC). PC 1 is dominated by the dark storm, providing most variance to the spectral map. The 2nd to 4th PCs show the ammonia ice S-shape feature. Each principal component map is normalised to unity for clarity.

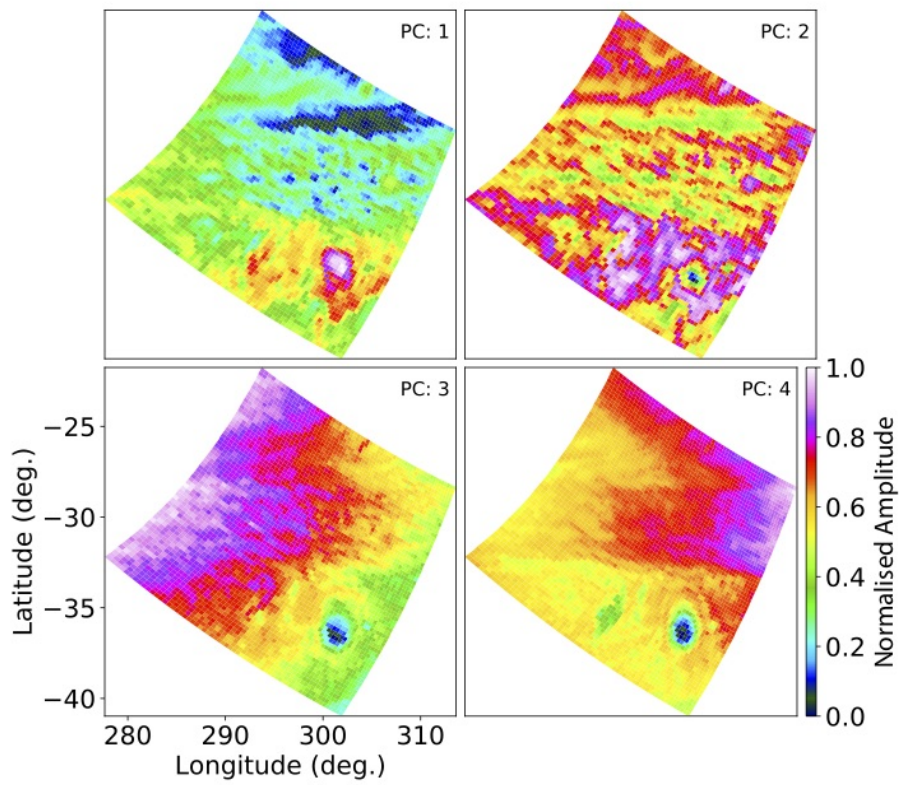


Figure 9: **Spatial map of the principal component analysis of the full wavelength range of the instrument.** Clearly visible are the dark storm, the extended SR feature and the high-altitude CH₄ clouds.

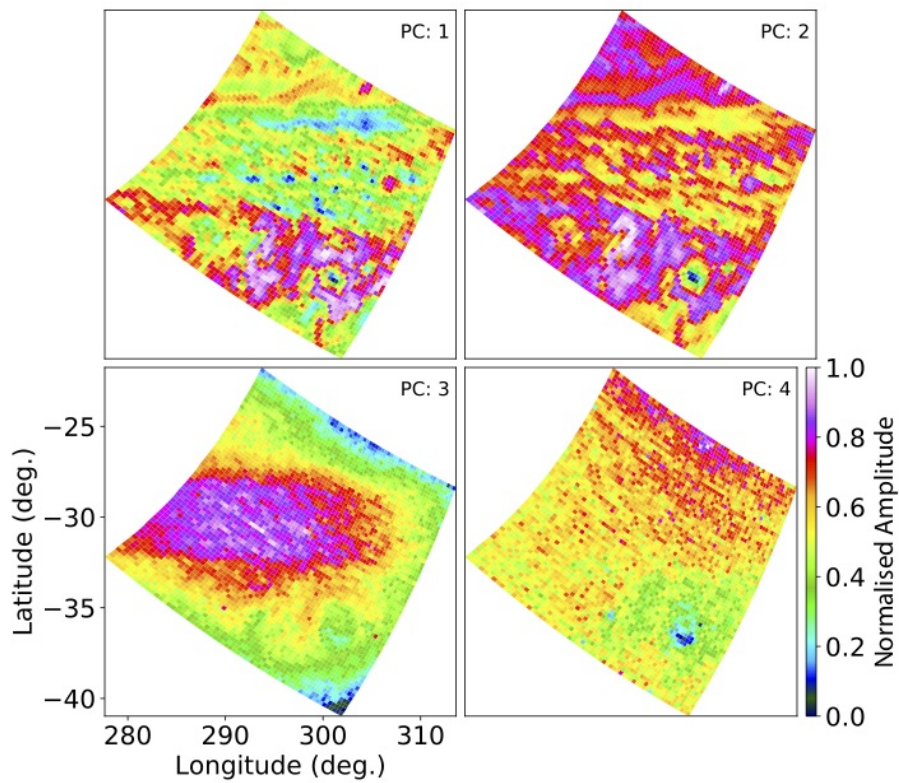


Figure 10: **Spatial map of the principal component analysis of the 4.0 - 5.1 μm wavelength range.** This range depicts the thermal emission component of the spectrum. PCs 1 & 2 show the storm (SR) region surrounding the dark storm.

Additional PlanetNet component maps

Additional maps showing individual components as recognised by PlanetNet. For more information, please refer to the main text.

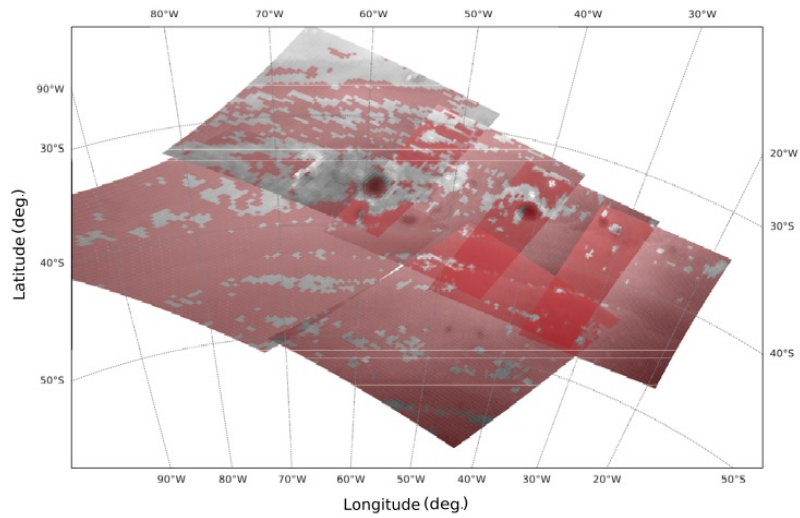


Figure 11: **Map of the second spectral/spatial feature identified by PlanetNet.** This feature corresponds to the orange shaded region (spectrum 2) in Figure 2 of the main text.

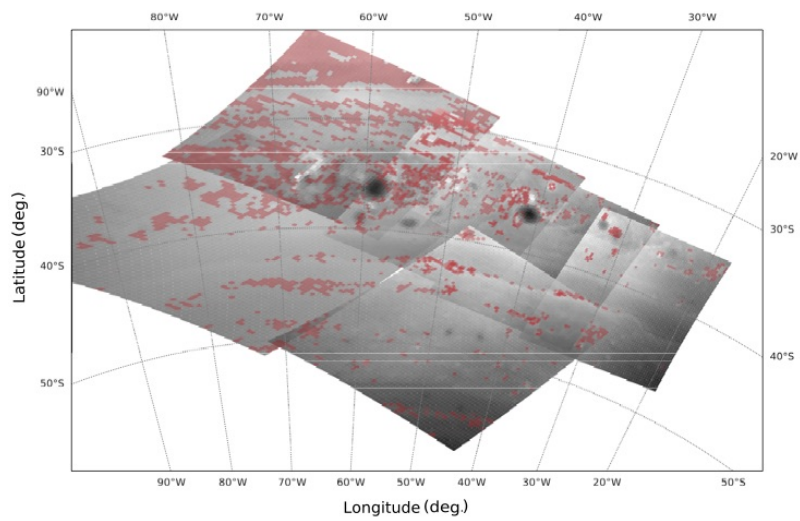


Figure 12: **Map of the third spectral/spatial feature identified by PlanetNet.** This feature corresponds to the red shaded region (spectrum 3) in Figure 2 of the main text.

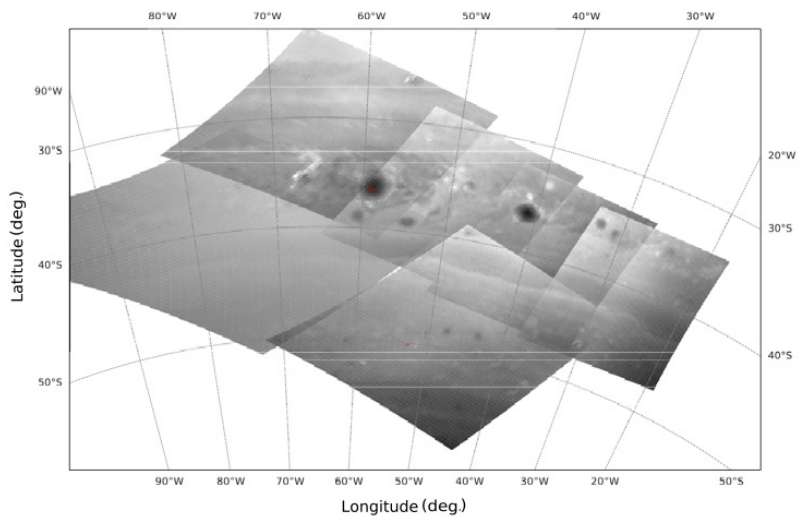


Figure 13: **Map of the fourth spectral/spatial feature identified by PlanetNet.** This feature corresponds to the green shaded region (spectrum 5) in Figure 2 of the main text.

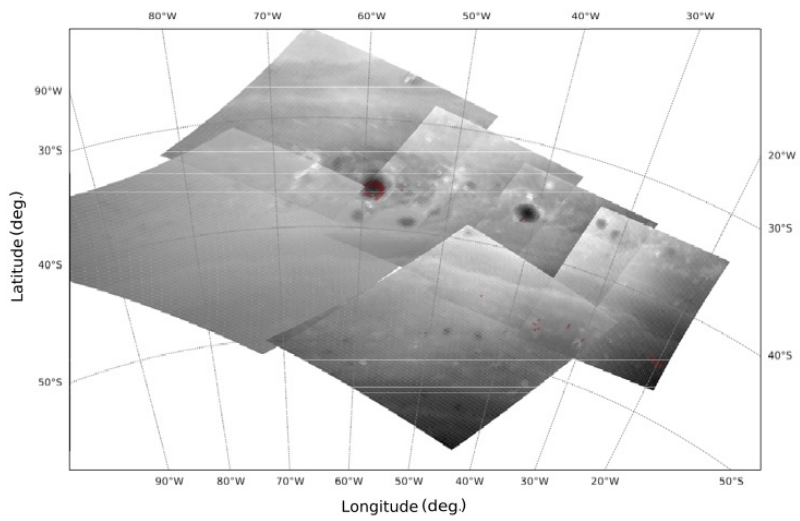


Figure 14: **Map of the fifth spectral/spatial feature identified by PlanetNet.** This feature corresponds to the purple shaded region (spectrum 6) in Figure 2 of the main text.

Training and validation

Additional plots and tables pertaining to the training and validation outlined in the Method section of the main text.

Table 1: **Summary of PlanetNet hyper-parameters and their values.**

Hyperparameters	Value
H^0, W^0	20,20
N^1, N^2	15,40
K_λ^l, K_ϕ^l	$4 \times 1, 4 \times 4$
P_ϕ, P_λ	$2 \times 2, 2 \times 1$
M^1, M^2	1024, 10

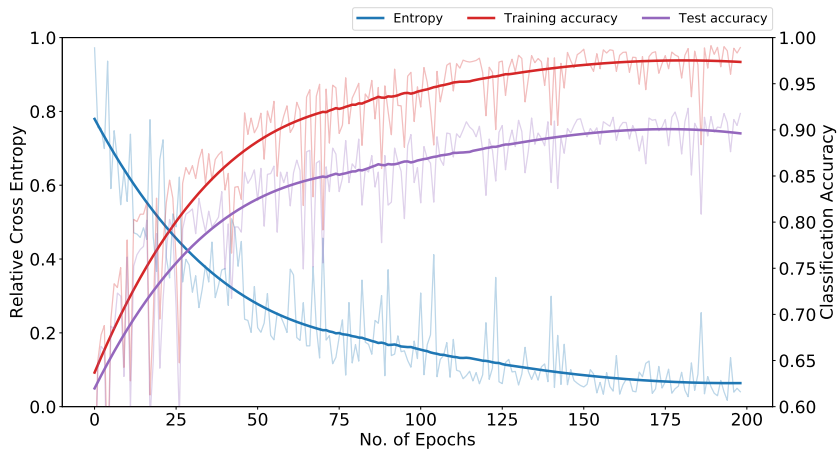


Figure 15: **PlanetNet accuracies as function of training duration.** Left Axis/Blue: Relative cross entropy as function of training duration. Right axis: Training (red) and Test (purple) set accuracy as function of training duration. Bold lines present a smoothed average.

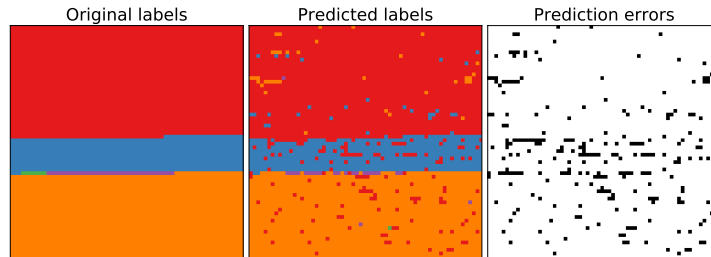


Figure 16: **PlanetNet recognition accuracy test on the re-sampled training cube (RTC)**. The RTC is shown in the left panel (Original Labels) with the PlanetNet predicted labels in the centre and the error-matrix (Prediction errors) showing the wrongly identified pixels. 94% of all pixels were correctly identified.

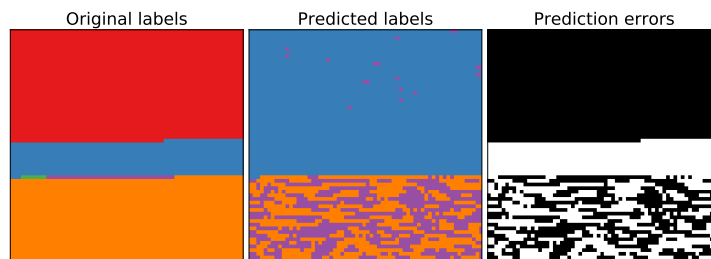


Figure 17: **PlanetNet recognition accuracy test on the re-sampled training cube (RTC) when only the spectral information was considered**. The layout is identical to supplementary figure 16 but here the spatial information was artificially set to zero, resulting in a very low recognition accuracy of 32%.

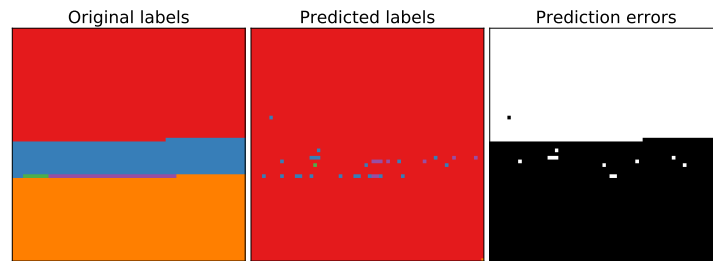


Figure 18: **PlanetNet recognition accuracy test on the re-sampled training cube (RTC) when only the spatial information was considered.** The layout is identical to supplementary figure 16 but here the spectral information was artificially set to zero, resulting in a very low recognition accuracy of 48%.